

**Semestr VI**

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:		Rozproszone Bazy Danych
Nazwa w języku angielskim:		Distributed Database
Język wykładowy:		polski
Kierunek studiów, dla którego przedmiot jest oferowany:		Informatyka
Jednostka realizująca:		Wydział Nauk Ścisłych i Przyrodniczych
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):		fakultatywny
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):		pierwszego stopnia
Rok studiów:		trzeci
Semestr:		szósty
Liczba punktów ECTS:		3
Imię i nazwisko koordynatora przedmiotu:		prof. dr hab. inż. Andrzej Barczak
Imię i nazwisko prowadzących zajęcia:		prof. dr hab. inż. Andrzej Barczak dr Anna Kołkowicz mgr Zbigniew Młynarski
Założenia i cele przedmiotu:		Celem kursu jest zapoznanie studentów z teoretycznymi zagadnieniami związanymi z rozproszonymi bazami danych, ich trendami rozwojowymi oraz modelami stosowanymi w przetwarzaniu rozproszonym
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	zna i rozumie podstawowe zagadnienia z zakresu teoretycznych podstaw rozproszonych baz danych, wykorzystywanych modeli oraz ich projektowania	K_W09
W_02	orientuje się w obecnym stanie oraz najnowszych trendach rozwojowych z zakresu rozproszonych baz danych, zna i rozumie podstawowe zagadnienia z zakresu standardów i norm z zakresu baz danych	K_W06
W_03	Zna i rozumie zagadnienia związane z podstawowymi modelami i problemami przetwarzania rozproszonego i równoległego, zna i rozumie modele systemów rozproszonych i techniki przetwarzania rozproszonego, w tym w szczególności w zakresie rozproszonych baz danych	K_W01, K_W07
Symbol efektu	Efekt uczenia się: UMIEJĘTNOŚCI	Symbol efektu kierunkowego
U_01	potrafi pozyskiwać informacje na temat rozproszonych baz danych z literatury i innych źródeł, w tym zwłaszcza internetowych; potrafi analizować, interpretować, porządkować, oceniać przydatność i użyteczność oraz agregować i integrować uzyskane informacje, a także wyciągać wnioski z ich treści i formułować opinie	K_U01

U_02	przy identyfikowaniu i formułowaniu specyfikacji zadań inżynierskich w dziedzinie baz danych oraz przy ich rozwiązywaniu potrafi dokonać wstępnej oceny proponowanych rozwiązań i podejmowanych działań inżynierskich	K_U09
U_03	potrafi zaprojektować, zaimplementować oraz przetestować system informatyczny o charakterze rozproszonej bazy danych	K_U09, K_U10, K_U12
U_04	potrafi ocenić przydatność dostępnych metod i narzędzi służących do rozwiązywania zadań inżynierskich w dziedzinie baz danych i rozwiązywać praktyczne zadania inżynierskie wymagające korzystania ze standardów i norm inżynierskich oraz stosowania technologii informatycznych, wykorzystując doświadczenie zdobyte w środowisku zajmującym się zawodowo działalnością inżynierską	K_U10, K_U24
<b>Symbol efektu</b>	<b>Efekt uczenia się: KOMPETENCJE SPOŁECZNE</b>	<b>Symbol efektu kierunkowego</b>
K_01	Jest gotów do podejmowania decyzji i krytycznej oceny własnych rozwiązań w rozwiązywaniu zadań projektowych z zakresu rozproszonych baz danych	K_K01
K_02	Jest gotów do uznania znaczenia wiedzy w rozwiązywaniu zadań projektowych z zakresu rozproszonych baz danych oraz krytycznie potrafi ocenić swoje działania	K_K01
<b>Forma i typy zajęć:</b>	studia stacjonarne: wykłady (21 godz.), ćwiczenia (24 godz.) studia niestacjonarne: wykłady (15 godz.), ćwiczenia (15 godz.)	
<b>Wymagania wstępne i dodatkowe:</b>		
Warunkiem uczestnictwa w zajęciach jest wcześniejsze uzyskanie zaliczenia z następujących przedmiotów: 1. Bazy danych 2. Podstawy przetwarzania równoległego 3. Platformy programowania 4. Systemy operacyjne. lub znajomość literatury obowiązującej w tych przedmiotach.		
<b>Treści modułu kształcenia:</b>		
<ol style="list-style-type: none"> <li><b>Wprowadzenie do systemów rozproszonych.</b> Podstawowe pojęcia związane z rozproszeniem. Rozproszenie zasobów. Główne aspekty rozproszenia baz danych. Przezroczystość rozproszenia. Współdziałanie. Heterogeniczność. <b>Definiowanie łącznika bazy danych.</b> *;</li> <li><b>Rozproszona baza danych - podstawowe pojęcia, cele i zalety rozproszenia.</b> Komunikacja: aplikacja - baza danych, dostęp do zbioru nazw usług. Lokalny zbiór nazw usług, katalogowa. Baza danych LDAP. Serwer nazw, adresowanie serwera. Zewnętrzny serwis katalogowy. Konfigurowanie lokalnego zbioru nazw usług. Konfigurowanie środowiska klienta. <b>Łącznik współdzielony. Perspektywy i synonimy.</b></li> <li><b>Typy i reguły rozproszonych baz danych.</b> Homogeniczność i heterogeniczność programowania. Stopień lokalnej autonomii. Federacyjny system baz danych i zarządzanie nim. Relacyjno-obiektowe bazy danych. Techniki sterowania współbieżnego. Reguły Date. Fragmentacja pozioma, pionowa i mieszana. Kryteria poprawności fragmentacji. <b>Transakcje rozproszone cz. 1.</b></li> <li><b>Architektura rozproszonych baz danych.</b> Architektura rozproszonej bazy danych. Specjalizowane oprogramowanie sieciowe. Łącznik bazy danych, perspektywa, synonim, migawka. Nazewnictwo baz danych w sieci, domena i nazwa globalna. Architektura klient -</li> </ol>		

serwer, mechanizm komunikacji między klientem a serwerem. Określenie jednostki komunikacji klient-serwer, funkcje po stronie klienta i po stronie serwera. Architektura klient - broker - serwer, broker - pośrednik w dostępie do odległych zasobów. Architektura odniesienia ANSI.

**Transakcje rozproszone cz. 2.**

**5. Federacyjne systemy baz danych.** Zarządzanie hierarchiami elementów baz danych. Protokół drzewiasty. Sterowanie współbieżnością za pomocą znaczników czasowych. Sterowanie współbieżnością za pomocą walidacji. Tryby integrowania danych.

**Protokół zatwierdzania dwufazowego.**

**6. Podejścia do projektowania rozproszonych baz danych.** Podejście top-down, bottom-up, ad-hoc. Podział schematu logicznego, metody. Podejście oparte o globalny schemat. Problem alokacji. Problematyka replikacji. Określanie jednostki replikacji, ilości replikowanych danych, momentu i sposobu odświeżania. **Awarie i odtwarzanie transakcji rozproszonych.**

**7. Komunikacja: aplikacja - baza danych.** Szeregi i plany szeregowane. Szeregowalność kolizyjna. Zapewnienie atomowości rozproszonej. Zatwierdzanie dwufazowe. Odtwarzanie transakcji rozproszonych. **Migawka – perspektywa zmaterializowana.**

**8. Przetwarzanie i optymalizacja zapytań rozproszonych.** Transakcja rozproszona. Architektura zarządzania transakcjami rozproszonymi. Protokół 2PC, scentralizowany, zdecentralizowany i liniowy. Rodzaje optymalizacji poleceń, wybór optymalizatora i celu optymalizacji. Generowanie statystyk i algorytmy łączenia tabel. Wykonywanie zapytań rozproszonych. Filtrowanie, grupowanie i sortowanie danych z tabeli zdalnej. Łączenie tabel, wykorzystanie wskazówek w łączeniu tabel. **Typy migawek. Dziennik migawki.**

**9. Replikacje.** Odświeżanie replik. Migawka - perspektywa zmaterializowana, typ migawki. Dziennik migawki, definiowanie dziennika. Implementacja dziennika, fizyczne parametry składowania dziennika. Modyfikowanie i usuwanie dziennika. Grupa odświeżania.

**Optymalizacja zapytań rozproszonych.**

**10. Przetwarzanie transakcji.** Szeregowalność i odtwarzalność. Szeregowalność perspektywiczna. Rozwiązywanie problemu zakleszczeń. Zatwierdzanie rozproszone. Transakcje o długim czasie trwania. **Grupy odświeżania.**

**11. Partycjonowanie tabel i indeksów.** Algorytmy partycjonowania danych, partycjonowanie tabel. Partycjonowanie bazujące na wartości, haszowe i hybrydowe. Fizyczne parametry składowania tabel partycjonowanych. Zarządzanie tabelami partycjonowanymi. Partycjonowanie indeksów, typy indeksów. Zarządzanie indeksami partycjonowanymi. **Partycjonowanie danych.**

\* Kursywą zaznaczono zagadnienia praktyczne realizowane w formie zajęć laboratoryjnych

**Literatura podstawowa:**

1. Bębel B., Wrembel R.; Oracle. Projektowanie rozproszonych baz danych; Wydawnictwo Helion, 2003
2. Wrembel R., Bębel B., Oracle : projektowanie rozproszonych baz danych : wiedza niezbędna do projektowania oraz zarządzania rozproszonymi bazami danych Wydawnictwo Helion, 2005
3. Andrzej Barczak, Dariusz Zacharczuk, Damian Pluta., Metody optymalizacji rozproszonych baz danych w systemie Oracle w Studia Informatica. - 2015, Vol. 36, no. 1 (119), s. 113-129.
4. Andrzej Barczak, Dariusz Zacharczuk, Damian Pluta., Methods of optimization of distributed databases in oracle – part 1 w Studia Informatica: Systems and Information Technology - 2016, nr 1-2 (20), s. 5-15
5. Andrzej Barczak, Dariusz Zacharczuk, Damian Pluta., Methods of optimization of distributed databases in oracle – part 1 w Studia Informatica: Systems and Information Technology - 2017, nr 1(21),

**Literatura dodatkowa:**

1. Garcia-Molina H., Ullman J. D., Widom J.; Systemy baz danych Pełny wykład; Wydawnictwa Naukowo Techniczne, 2006
2. Mueller J. P.; Poznaj Soap; Wydawnictwo Mikom, 2002

3. Di Stefano M.; Distributed Data Management For Grid Computing; John Wiley & Sons, Inc., 2005
4. Durbin J., Creekbaum W., Bobrowski S., Vasterd P.; Oracle8i Distributed Database Systems; Oracle Corporation. 1999
5. Tyagi S., McCammon K., Vorburger M., Bobzin H.; Java Data Objects; Wydawnictwo Helion, 2004
6. Connolly T., Begg C.; Systemy baz danych - Praktyczne metody projektowania, implementacji i zarządzania. Tom 1, 2, Wydawnictwo RM,
7. Elmasri R., Navathe S. B., Wprowadzenie do systemów baz danych;

### Planowane formy/działania/metody dydaktyczne:

Wykład tradycyjny wspomagany technikami multimedialnymi, laboratoria – praktyczna praca na komputerze. Zamieszczanie na stronach internetowych zagadnień teoretycznych i zadań ćwiczeniowych.

### Sposoby weryfikacji efektów kształcenia osiągniętych przez studenta:

Efekty W\_01 i W\_03 będą weryfikowane na kolokwium. Przykładowe pytania

- *Wyjaśnij podstawowe pojęcia: przezroczystość rozproszenia, współdziałanie, heterogeniczność itd.*
- *Omów główne aspekty rozproszenia bazy danych*
- *Wymień i scharakteryzuj typy rozproszonych baz danych*
- *Omów architekturę systemu rozproszonej bazy danych,*
- *Omów wady i zalety rozproszonych baz danych*

Efekty U\_01 - U\_04 weryfikowane będą w trakcie zajęć oraz sprawdzane na kolokwium. Przykładowe zadania:

- *Przeprowadź optymalizację zapytań w systemie rozproszonej bazy danych.*
- *Realizacja transakcji rozproszonych, problemy związane z przetwarzaniem rozproszonym,*
- *Przygotuj plan wykonania transakcji rozproszonej,*
- *Przeprowadź odświeżanie przyrostowe migawki*

Efekty K\_01 i K\_02 będą weryfikowane, w oparciu o posiadaną wiedzę i umiejętności, w czasie zajęć laboratoryjnych, podczas zaliczania zadania indywidualnego, a także będą sprawdzane na kolokwium.

Przykładowe zadania:

- *Zaprojektuj strukturę rozmów z potencjalnymi użytkownikami rozproszonej bazy danych w celu sformułowania podstawowych wymagań funkcjonalnych na system,*

Przed egzaminem studenci będą mieli dostęp do pełnej listy pytań.

Tematy zadań indywidualnych podawane są studentom najpóźniej na 3 zajęciach laboratoryjnych.

Zadania na następne laboratorium muszą być dostępne co najmniej tydzień przed zajęciami. Student, na podstawie podanej literatury, musi się do nich przygotować samodzielnie lub korzystając z konsultacji.

### Forma i warunki zaliczenia:

Moduł kończy się zaliczeniem z oceną. Ocena końcowa jest wystawiana na podstawie zajęć laboratoryjnych i jednego kolokwium pisemnego przeprowadzonego na ostatnim wykładzie. Na zaliczenie laboratorium składają się oceny cząstkowe uzyskane na regularnych zajęciach z nauczycielem akademickim oraz z samodzielnie wykonanego zadania indywidualnego według schematu:

- Regularne zajęcia – 40 pkt.,
- Obrona zadania indywidualnego – 20 pkt.

Zajęcia laboratoryjne będą zaliczone w wypadku uzyskania co najmniej połowy punktów z poszczególnych form aktywności studenta: regularne zajęcia – co najmniej 20 pkt., obrona indywidualnego zadania – co najmniej 10 pkt. Na tej formie zajęć student może maksymalnie uzyskać 60 pkt.

Za pisemne kolokwium można na nim uzyskać do 40 pkt. Zaliczenie kolokwium jest możliwe po uzyskaniu co najmniej 20 pkt. Ocena końcowa z modułu (po zaliczeniu wszystkich części składowych), w zależności od sumy uzyskanych punktów (maksymalnie 100pkt.) jest następująca (w nawiasach ocena wg skali ECTS):

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

Poprawy: Trzy terminy egzaminu pisemnego. Drugi termin zaliczenia laboratorium – w toku sesji egzaminacyjnej.

Bilans punktów ECTS*:	
Studia stacjonarne	
Aktywność	Obciążenie studenta
Udział w wykładach	21 godz.
Udział w ćwiczeniach laboratoryjnych	24 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	15 godz.
Udział w konsultacjach godz. z przedmiotu	5 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	10 godz.
Sumaryczne obciążenie pracą studenta	75 godz.
Punkty ECTS za przedmiot	3 ECTS
Studia niestacjonarne	
Aktywność	Obciążenie studenta
Udział w wykładach	15 godz.
Udział w ćwiczeniach laboratoryjnych	15 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	28 godz.
Udział w konsultacjach godz. z przedmiotu	2 godz.

Przygotowanie się do egzaminu i obecność na egzaminie	15 godz.
Sumaryczne obciążenie pracą studenta	75 godz.
Punkty ECTS za przedmiot	3 ECTS

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:		Środowiska Programowania Aplikacji Wirtualnych i Multimedialnych
Nazwa w języku angielskim:	Virtual and Multimedia Application Programming Environment	
Język wykładowy:	polski	
Kierunek studiów, dla którego przedmiot jest oferowany:	informatyka	
Jednostka realizująca:	Wydział Nauk Ścisłych i Przyrodniczych	
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):	fakultatywny	
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):	pierwszego stopnia	
Rok studiów:	trzeci	
Semestr:	szósty	
Liczba punktów ECTS:	3	
Imię i nazwisko koordynatora przedmiotu:	dr Grzegorz Terlikowski	
Imię i nazwisko prowadzących zajęcia:	dr Grzegorz Terlikowski	
Założenia i cele przedmiotu:	Celem przedmiotu jest omówienie zagadnień związanych z aplikacjami wirtualnymi i multimedialnymi, w szczególności z technologiami, językami programowania i środowiskami	
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	Zna i rozumie na czym polega wirtualizacja oraz zna założenia dotyczące programowania aplikacji wirtualnych i multimedialnych.	K_W03,
W_02	Zna i rozumie funkcjonowanie narzędzi tj: IntelliJ IDEA, Visual Studio oraz frameworków takich jak: WPF, Ionic, React Native.	K_W10, K_W12
W_03	Zna i rozumie różne atrybuty multimedii i wie jak je wykorzystać w tworzonych przez siebie aplikacjach. Rozumie potrzebę wykorzystania elementów sieciowych w aplikacjach multimedialnych. Zna i rozumie przydatność zastosowań aplikacji wirtualnych i multimedialnych.	K_W03, K_W10, K_W12
W_04	Zna i rozumie zasady tworzenia aplikacji w językach: C#, Java, XAML, HTML5. Zna i rozumie działanie protokołów umożliwiających transfer danych multimedialnych	K_W06, K_W10, K_W12
Symbol efektu	Efekt uczenia się: UMIEJĘTNOŚCI	Symbol efektu kierunkowego



U_01	Potrafi w praktyce wykorzystać narzędzia umożliwiające tworzenie aplikacji wirtualnych i multimedialnych. Potrafi wesprzeć się adekwatnymi materiałami dydaktycznymi. Potrafi posługiwać się pojęciami związanymi z multimediami i wirtualizacją.	K_U01, K_U10, K_U11
U_02	Potrafi implementować aplikacje multimedialne w technologiach: WPF, Ionic, React Native. Potrafi tworzyć bogate interfejsy użytkownika(Rich Internet Application, RIA).	K_U10, K_U11
U_03	Potrafi implementować aplikacje hybrydowe wykorzystując technologie webowe oraz przetestować poprawność ich działania.	K_U10, K_U11, K_U12
Symbol efektu	<b>Efekt uczenia się: KOMPETENCJE SPOŁECZNE</b>	Symbol efektu kierunkowego
K_01	Jest gotów do podejmowania decyzji i krytycznej oceny własnych rozwiązań w rozwiązywaniu zadań projektowych z zakresu implementacji aplikacji multimedialnych	K_K01
K_02	Jest gotów do uznania znaczenia wiedzy w rozwiązywaniu zadań projektowych z zakresu aplikacji wirtualnych i multimedialnych.	K_K01
<b>Forma i typy zajęć:</b>	studia stacjonarne: wykłady (21 godz.), ćwiczenia laboratoryjne (24 godz.) studia niestacjonarne: wykłady (15 godz.), ćwiczenia laboratoryjne (15 godz.)	
<b>Wymagania wstępne i dodatkowe:</b>		
Umiejętność wykorzystania języków: HTML(5), Java Script, Type Script, C#.		
<b>Treści modułu kształcenia:</b>		
<ol style="list-style-type: none"> <li><b>Wirtualizacja a Multimedia.</b> Co to jest wirtualizacja? Zalety, wady i zastosowanie podstawowych typów wirtualizacji. Przegląd technologii do wytwarzania aplikacji multimedialnych.</li> <li><b>Wirtualizacja aplikacji</b> – porównanie narzędzi służących do wirtualizacji aplikacji Microsoft Application Virtualization, Symantec Workspace Streaming, Endeavors Jukebox Application.</li> <li><b>Docker</b> – nowoczesne podejście do projektowania i wdrażania aplikacji.</li> <li><b>Wprowadzenie do WPF.</b> Uruchamianie przykładowych aplikacji, Instalowanie przykładowych aplikacji. Zapoznanie się z językiem XAML. Implementacja prostej aplikacji w WPF.</li> <li><b>Animacje w WPF.</b> Tworzenie scenariuszy animacji. Proste animacje, Animacje z klatkami kluczowymi, Wyzwalacze i Behawiory.</li> <li><b>Rola silników przeglądarek w tworzeniu aplikacji hybrydowych i wieloplatformowych.</b> Tworzenie aplikacji w WPF z wykorzystaniem silnika przeglądarki. Wsparcie dla multimediiów w HTML 5 i CSS3.</li> <li><b>Multimedialne mapy na stronach WWW.</b> Google Maps, OpenStreetMap i inne. Poligony, znaczniki, opisy, warstwy,</li> <li><b>Przegląd rozwiązań wieloplatformowych.</b> Podejścia do tworzenia multimedialnych aplikacji wieloplatformowych. Xamarin, Phone Gap, Vue.js i inne.</li> <li><b>Wieloplatformowe aplikacje mobilne.</b> Ionic jako przykłady technologii umożliwiających tworzenie wieloplatformowych aplikacji mobilnych wykorzystujących podejście hybrydowe.</li> <li><b>Wieloplatformowe aplikacje mobilne c.d.</b> React Native jako kolejna technologia umożliwiająca tworzenie wieloplatformowych aplikacji mobilnych (alternatywa do Ionic).</li> <li><b>Protokoły wspierające transmisje multimediiów.</b> Omówienie WebSockets. Protokoły RTP, RTTP, RTMP, RTCP, RTSP.</li> </ol>		
<b>Literatura podstawowa:</b>		

1. R. McKendrick, S. Gallagher. Docker. Programowanie aplikacji dla zaawansowanych. Wydanie II. Wydawnictwo: Helion, 2018 .
2. J. Matulewski, MVVM i XAML w Visual Studio 2015(ebook), Helion 2016.
3. T. Parisi. Aplikacje 3D. Przewodnik po HTML5, WebGL i CSS3. Helion 2014.

#### **Literatura dodatkowa:**

1. Gaurav Saini. Hybrid Mobile Development with Ionic (ebook). Wydawnictwo: Packt Publishing 2017.
2. Eric Masiello, Jacob Friedmann. Mastering React Native (ebook). Wydawnictwo: Packt Publishing 2017.
3. G. Taskos. Xamarin. Tworzenie aplikacji cross-platform. Receptury. Wydawnictwo: Helion 2017.

#### **Planowane formy/działania/metody dydaktyczne:**

Wykład tradycyjny wspomagany technikami multimedialnymi, ćwiczenia laboratoryjne wspomagane technikami multimedialnymi. Zamieszczanie na stronach internetowych problemów i zadań laboratoryjnych.

#### **Sposoby weryfikacji efektów kształcenia osiągniętych przez studenta:**

Efekty W\_01 – W\_03 będą sprawdzane podczas pisemnego zaliczenia wykładu na ocenę. Część pytań może mieć charakter otwarty, a część zamknięty (test).

Przykładowe pytania o charakterze otwartym:

- Wymień i scharakteryzuj znane Ci typy wirtualizacji.
- Scharakteryzuj rolę webowych technologii w tworzeniu aplikacji RIA.

Przykładowe pytania o charakterze zamkniętym:

1. Który kontener WPF umożliwi pozycjonowanie absolutne?
  - a. StackPanel,
  - b. Canvas,
  - c. StackPanel,
2. Który język znacznikowy jest dostępny na platformie WPF.
  - a. MXML,
  - b. XAML,
  - c. FXML.

Efekt U\_01 - U\_03 oraz K1 - K2 będą systematycznie sprawdzane na zajęciach laboratoryjnych.

Zadania na następne laboratorium muszą być dostępne co najmniej tydzień przed zajęciami. Student, na podstawie podanej literatury, musi się do nich samodzielnie lub korzystając z konsultacji przygotować.

Przykładowe zadanie:

- Wykorzystując silnik GeckoFX, zaimplementuj prostą aplikację hybrydową – formularz rejestracyjny. Użyj animacji CSS3 w celu podkreślenia multimedialnego charakteru aplikacji.

#### **Forma i warunki zaliczenia:**

Moduł kończy się zaliczeniem z oceną. Ocena końcowa jest wystawiana na podstawie zajęć laboratoryjnych i jednego kolokwium pisemnego przeprowadzonego na ostatnim wykładzie. Na zaliczenie laboratorium składają się oceny częściowe uzyskane na regularnych zajęciach z nauczycielem akademickim oraz z samodzielnie wykonanego zadania indywidualnego według schematu:

- Regularne zajęcia – 39 pkt.,
- Obrona zadania indywidualnego – 21 pkt.

Zajęcia laboratoryjne będą zaliczone w wypadku uzyskania co najmniej połowy punktów z poszczególnych form aktywności studenta: regularne zajęcia – co najmniej 20 pkt., obrona indywidualnego zadania – co najmniej 10 pkt. Na tej formie zajęć student może maksymalnie uzyskać 60 pkt.

Za pisemne kolokwium można uzyskać do 40 pkt. Zaliczenie kolokwium jest możliwe po uzyskaniu co najmniej 20 pkt. Ocena końcowa z modułu (po zaliczeniu wszystkich części składowych), w zależności od sumy uzyskanych punktów (maksymalnie 100pkt.) jest następująca (w nawiasach ocena wg skali ECTS):

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

Bilans punktów ECTS\*:

#### Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	21 godz.
Udział w ćwiczeniach laboratoryjnych	24 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	15 godz.
Udział w konsultacjach godz. z przedmiotu	5 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	10 godz.
Sumaryczne obciążenie pracą studenta	75 godz.
Punkty ECTS za przedmiot	3 ECTS

#### Studia niestacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	15 godz.
Udział w ćwiczeniach laboratoryjnych	15 godz.

Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	28 godz.
Udział w konsultacjach godz. z przedmiotu	2 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	15 godz.
Sumaryczne obciążenie pracą studenta	75 godz.
Punkty ECTS za przedmiot	3 ECTS

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:		Seminarium dyplomowe I
Nazwa w języku angielskim:	diploma seminar	
Język wykładowy:	polski	
Kierunek studiów, dla którego przedmiot jest oferowany:		informatyka
Jednostka realizująca:	Wydział Nauk Ścisłych i Przyrodniczych	
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):		obowiązkowy
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):		pierwszego stopnia
Rok studiów:	Trzeci	
Semestr:	szósty	
Liczba punktów ECTS:	1	
Imię i nazwisko koordynatora przedmiotu:		prof. dr hab. inż. Andrzej Barczak
Imię i nazwisko prowadzących zajęcia:		prof. dr hab. inż. Andrzej Barczak dr hab. Krzysztof Szkatuła
Założenia i cele przedmiotu:		Założono, że po kursie studenci będą znali podstawowe zasady związane z etapami pisania pracy. Celem kursu jest zapoznanie studentów z podstawowymi zasadami pisania prac inżynierskich
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	zna i rozumie cywilizacyjne znaczenie informatyki, zna i rozumie jej rolę w życiu społeczeństwa oraz zagrożenia związane z jej zastosowaniami	K_W03
Symbol efektu	Efekt uczenia się: UMIEJĘTNOŚCI	Symbol efektu kierunkowego
U_01	potrafi pozyskiwać informacje z literatury, baz danych i innych, w tym zwłaszcza internetowych źródeł; potrafi analizować, interpretować oraz integrować uzyskane informacje, a także oceniać ich użyteczność w aspekcie wykonywanej pracy kwalifikacyjnej, wyciągać wnioski oraz formułować i uzasadniać opinie	K_U01
K_U02	potrafi dostrzegać aspekty pozatechniczne, w tym środowiskowe, ekonomiczne i prawne przy projektowaniu, implementowaniu i wdrażaniu systemów informatycznych	K_U23
Symbol efektu	Efekt uczenia się: KOMPETENCJE SPOŁECZNE	Symbol efektu kierunkowego

<b>K_01</b>	jest gotów do odpowiedzialnego pełnienia roli zawodowej informatyka, w tym do przestrzegania zasad etyki zawodowej i wymagania tego od innych i do dbania o dorobek i tradycję zawodu informatyka	<b>K_K04</b>
<b>K_K02</b>	jest gotów do krytycznej oceny posiadanej wiedzy oraz jest gotów do uznawania znaczenia wiedzy w rozwiązywaniu problemów poznawczych i praktycznych z zakresu informatyki	<b>K_K01</b>
<b>Forma i typy zajęć:</b>	Studia stacjonarne: seminarium 15 godz. Studia niestacjonarne: seminarium 10 godz.	
<b>Wymagania wstępne i dodatkowe:</b>		
Brak		
<b>Treści modułu kształcenia:</b>		
Treści seminarium: <ol style="list-style-type: none"> <li>1. Prezentacja zagadnień związanych z pisaniem pracy promocyjnej.</li> <li>2. Weryfikacja i zatwierdzenie tematu, planu i harmonogramu pracy dyplomowej.</li> <li>3. Przygotowanie prezentacji poświęconej przyszłej pracy dyplomowej zatwierdzonej przez potencjalnego promotora</li> <li>4. Wygłoszenie i obrona prezentacji.</li> </ol>		
<b>Literatura podstawowa:</b>		
<ol style="list-style-type: none"> <li>1. Sobaniec C.: Jak pisać pracę inżynierską/magisterską. <a href="http://www.cs.put.poznan.pl/sobaniec/edu/jak_pisacmgr.pdf">www.cs.put.poznan.pl/sobaniec/edu/jak_pisacmgr.pdf</a></li> <li>2. Starecki T.: Praca dyplomowa – jak realizować, jak pisać i dlaczego. <a href="http://www.ise.pw.edu.pl/impuls/Dyplom.pdf">www.ise.pw.edu.pl/impuls/Dyplom.pdf</a></li> <li>3. Opoka E.: Uwagi o pisaniu i redagowaniu prac dyplomowych na studiach technicznych. Wyd. Politechnika Śląska, Gliwice 2001</li> <li>4. Wytrębowski J.: O poprawności językowej publikacji naukowo-technicznych. w: Zagadnienia Naukoznawstwa, Nr 1(179) 2009</li> </ol>		
<b>Literatura dodatkowa:</b>		
<ol style="list-style-type: none"> <li>1. Kwaśniewski A.: Jak pisać pracę dyplomową. <a href="http://zpt2.tele.pw.edu.pl/~andrzej/TP/wyklad/wyklad-pdf/TP-praca_dypl.pdf">http://zpt2.tele.pw.edu.pl/~andrzej/TP/wyklad/wyklad-pdf/TP-praca_dypl.pdf</a></li> <li>2. Drozdowski M. Jak pisać prace dyplomową/magisterską. <a href="http://www.cs.put.poznan.pl/mdrozdowski/dyd/txt/jak_mgr.html">http://www.cs.put.poznan.pl/mdrozdowski/dyd/txt/jak_mgr.html</a>.</li> </ol>		
<b>Planowane formy/działania/metody dydaktyczne:</b>		
Zajęcia o charakterze seminaryjnym.		
<b>Sposoby weryfikacji efektów kształcenia osiągniętych przez studenta:</b>		
Efekty uczenia się W_01, U_01, U_02 oraz K_01 i K_K01 weryfikowane będą w toku zajęć seminaryjnych na podstawie przygotowania, udziału i aktywności poszczególnych studentów w zajęciach.		
<b>Forma i warunki zaliczenia:</b>		
Moduł podlega zaliczeniu (bez oceny). Podstawą zaliczenia jest przygotowanie prezentacji dotyczącej opracowywanej pracy inżynierskiej		
Bilans punktów ECTS:		

Studia stacjonarne	
Aktywność	Obciążenie studenta
Udział w seminarium	15 godz.
Samodzielne przygotowanie się do seminarium	8 godz.
Udział w konsultacjach godz. z przedmiotu	2 godz.
Sumaryczne obciążenie pracą studenta	25 godz.
Punkty ECTS za przedmiot	1 ECTS
Studia niestacjonarne	
Aktywność	Obciążenie studenta
Udział w seminarium	10 godz.
Samodzielne przygotowanie się do seminarium	14 godz.
Udział w konsultacjach godz. z przedmiotu	1 godz.
Sumaryczne obciążenie pracą studenta	25 godz.
Punkty ECTS za przedmiot	1 ECTS

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:	Praktyka zawodowa (III)	
Nazwa w języku angielskim:	Apprenticeship III	
Język wykładowy:	polski	
Kierunek studiów, dla którego przedmiot jest oferowany:	Informatyka	
Jednostka realizująca:	Wydział Nauk Ścisłych i Przyrodniczych	
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):	Obowiązkowy	
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):	Pierwszego stopnia	
Rok studiów:	Trzeci	
Semestr:	Szósty	
Liczba punktów ECTS:	10	
Imię i nazwisko koordynatora przedmiotu:	Dr Artur Niewiadomski	
Imię i nazwisko prowadzących zajęcia:	Osoba delegowana z firmy/institucji	
Założenia i cele przedmiotu:	<p>Cele praktyki:</p> <ul style="list-style-type: none"> <li>• pogłębienie umiejętności zawodowych studenta</li> <li>• pogłębienie specjalistycznej wiedzy, umiejętności i kompetencji związanych z funkcjonowaniem firmy/institucji w zakresie stosowanych systemów informatycznych oraz rozwoju istniejących systemów i wytwarzania nowych aplikacji</li> <li>• pogłębienie i wykorzystanie w praktyce wiedzy i umiejętności nabytych podczas nauki oraz poprzednich etapów praktyki zawodowej, zwłaszcza tych związanych z wybraną specjalnością</li> <li>• zdobycie wiedzy i umiejętności związanych ze sposobami organizacji pracy indywidualnej i zespołowej</li> <li>• nawiązanie kontaktów zawodowych ułatwiających poszukiwanie pracy, firmy/institucji do odbycia kolejnych etapów praktyk, czy realizacji prac dyplomowych.</li> </ul>	
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego



W_01	Posiada wiedzę w zakresie organizacji i zarządzania czasem oraz budowy harmonogramu pracy indywidualnej i zespołowej.	K_W13
W_02	Zdobywa i pogłębia specjalistyczną wiedzę dziedzinową związaną z systemami i narzędziami informatycznymi wykorzystywanymi w firmie.	K_W13
W_03	Zna i rozumie w zaawansowanym stopniu zagadnienia związane z wytwarzaniem oprogramowania oraz administrowaniem systemami informatycznymi.	K_W13
W_04	Student orientuje się w potrzebach rynku pracy. Zna relację między wymaganiami pracodawców a wiedzą zdobytą w trakcie zajęć.	K_W13
<b>Symbol efektu</b>	<b>Efekt uczenia się: UMIEJĘTNOŚCI</b>	<b>Symbol efektu kierunkowego</b>
U_01	Potrafi planować czas pracy, nadawać priorytety zadaniom i je terminowo realizować. Potrafi wykorzystać wiedzę teoretyczną, praktyczną oraz narzędzia informatyczne do realizacji postawionych zadań.	K_U23, K_U24
U_02	Potrafi zadbać o własny wizerunek zawodowy. Potrafi zidentyfikować kierunki dalszego rozwoju na podstawie pozyskanej wiedzy, umiejętności oraz doświadczeń zawodowych. Potrafi nawiązywać i utrzymywać kontakty zawodowe.	K_U01
U_03	Potrafi wykorzystać w praktyce wiedzę i umiejętności związane z realizowaną specjalnością. Potrafi w praktyce stosować zasady bezpieczeństwa i ergonomii pracy.	K_U14, K_U23, K_U25
<b>Symbol efektu</b>	<b>Efekt uczenia się: KOMPETENCJE SPOŁECZNE</b>	<b>Symbol efektu kierunkowego</b>
K_01	Jest gotów do odpowiedzialnego pełnienia roli zawodowej informatyka, w tym do przestrzegania zasad etyki zawodowej i wymagania tego od innych i do dbania o dorobek i tradycję zawodu informatyka.	K_K04
K_02	Krytycznie ocenia działania własne, działania zespołów, którymi kieruje i organizacji w której uczestniczy. Potrafi dokonać samooceny własnych kompetencji i doskonalenia swoich kwalifikacji zawodowych.	K_K01
<b>Forma i typy zajęć:</b>	Praktyka (320 godzin)	
<b>Wymagania wstępne i dodatkowe:</b>		
<ol style="list-style-type: none"> <li>1. Podstawowa wiedza z zakresu funkcjonowania, projektowania i implementacji systemów informatycznych.</li> <li>2. Wiedza, umiejętności i kompetencje społeczne nabyte podczas praktyki po pierwszym i drugim roku studiów.</li> </ol>		

**Treści modułu kształcenia:**

1. Poznanie zasad bezpieczeństwa i higieny pracy na danym stanowisku oraz uwarunkowania prawne i etyczne stosownie do wykonywanych obowiązków.
2. Poznanie specyfiki działania przedsiębiorstwa, w którym jest odbywana praktyka.
3. Rozpoznanie obszarów działalności firmy wspomaganych komputerowo.
4. Zapoznanie z systemami i narzędziami informatycznymi wspomagającymi działalność firmy, a zwłaszcza wspierającymi zarządzanie i produkcję. W szczególności należy:
  - a. zapoznać się z dokumentacją techniczną sprzętu i oprogramowania,
  - b. rozpoznawać i rozwiązywać problemy związane z eksploatacją sprzętu i oprogramowania,
  - c. studiować możliwości optymalizacji, rozbudowy i modyfikacji infrastruktury teleinformatycznej, zgodnie z aktualnymi tendencjami rozwojowymi.
5. Dokonać oceny istniejącej infrastruktury i wykorzystywanych technologii informatycznych w przedsiębiorstwie pod kątem zgodności ze standardami oraz możliwości rozwoju i współpracy z innymi rozwiązaniami.
6. Ocenić aktualny stan oraz przyszłe potrzeby systemów informatycznych.
7. Współdział w projektowaniu nowych i ulepszaniu istniejących systemów informatycznych, biorąc pod uwagę:
  - a. wymagania i cele stawiane przed systemem informatycznym,
  - b. politykę bezpieczeństwa oraz procedury organizacyjne dotyczące wykorzystania infrastruktury informatycznej,
  - c. napotymane ograniczenia techniczne i biznesowe,
  - d. zagadnienia związane ze zwrotem kosztów inwestycji.
8. Realizacja zadań związanych z kierunkiem informatyka, a w szczególności ze specjalnością wybraną przez studenta odbywającego praktyki.
9. Prowadzenie dokumentacji przebiegu praktyk.

**Literatura podstawowa:**

Według zalecenia w miejscu odbywania praktyki.

**Literatura dodatkowa:**

Regulamin praktyk

**Planowane formy/działania/metody dydaktyczne:**

Cykl spotkań informacyjnych odnośnie celów i zakresu praktyki, wymaganych dokumentów i terminów oraz indywidualne konsultacje.

**Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:**

Wrywkowa hospitacja w miejscu praktyki, rozmowa ze studentem, ocena przedstawionej dokumentacji.

**Forma i warunki zaliczenia:**

Podstawą zaliczenia modułu jest ocena wystawiona studentowi w instytucji przyjmującej na praktykę i weryfikowana przez opiekuna praktyk na podstawie rozmowy lub arkusza hospitacyjnego. Ocena ta obejmuje efekty wykonania przydzielonych zadań, jak również sposób organizacji pracy i podejmowane działania (0-50pkt). Ponadto oceniana jest dokumentacja praktyk zarówno pod kątem merytorycznym jak i formalnym (m.in. kompletność dokumentacji, dotrzymywanie terminów; 0-50pkt).

Ocena końcowa z przedmiotu, w zależności od sumy uzyskanych punktów (maksymalnie 100 pkt) jest następująca (w nawiasach ocena wg skali ECTS):

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

**Bilans punktów ECTS:**

Studia stacjonarne i niestacjonarne

Aktywność	Obciążenie studenta
Udział w zorganizowanej formie pracy na terenie zakładu pracy – miejscu odbywania stażu	320 godzin
Sumaryczne obciążenie pracą studenta	320 godzin
Punkty ECTS za przedmiot	<b>10</b>

**Specjalność: Programowanie systemów i baz danych, semestr VI**

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:		Programowanie zaawansowane
Nazwa w języku angielskim:		Advanced programming
Język wykładowy:	polski	
Kierunek studiów, dla którego przedmiot jest oferowany:		Informatyka
Jednostka realizująca:	Wydział Nauk Ścisłych i Przyrodniczych	
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):		obowiązkowy
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):		pierwszego stopnia
Rok studiów:	trzeci	
Semestr:	szósty	
Liczba punktów ECTS:	5	
Imię i nazwisko koordynatora przedmiotu:		Waldemar Bartyna
Imię i nazwisko prowadzących zajęcia:		Waldemar Bartyna
Założenia i cele przedmiotu:		<p>Studenci przystępujący do tego przedmiotu powinni znać zasady programowania obiektowego, podstawy języka C# i wzorzec MVC.</p> <p>Celem przedmiotu jest zapoznanie studentów z projektowaniem i implementacją aplikacji wykorzystujących zaawansowane konstrukcje i mechanizmy programistyczne dostępne w języku C# na platformie .NET. Studenci będą dokładnie rozumieć znaczenie i sposób działania takich konstrukcji i mechanizmów jak obsługa wyjątków, polimorfizm, budowanie bazy wspólnego kodu, delegaty, refleksja, serializacja, LINQ i innych, oraz będą potrafili prawidłowo je wykorzystywać w procesie tworzenia aplikacji różnego typu.</p>
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	Zna i rozumie zagadnienia dotyczące zasad programowania obiektowego, między innymi, hermetyzacji, dziedziczenia i polimorfizmu.	K_W06

<b>W_02</b>	Zna i rozumie zaawansowane mechanizmy programistyczne, w tym: strukturalną obsługę wyjątków, interfejsy; generyczność, refleksję i serializację.	<b>K_W06</b>
<b>W_03</b>	Zna i rozumie elementy składające się na platformę .NET, język programowania C# i jego zaawansowane konstrukcje w tym: właściwości automatyczne, metody rozszerzeniowe, niejawne typowanie.	<b>K_W06</b>
<b>W_04</b>	Zna i rozumie sposoby komunikowania się między obiektami poprzez interfejsy i silnie typowane wskaźniki do metod, czyli delegaty.	<b>K_W06</b>
<b>W_05</b>	Zna i rozumie różne sposoby utrwalania danych (poprzez bazy danych, pliki XML), oraz sposoby korzystania z tych źródeł danych (ADO.NET i LINQ to SQL).	<b>K_W09</b>
<b>W_06</b>	Zna i rozumie zasady korzystania i mechanizmy związane z językiem zapytań zintegrowanym z językiem programowania (LINQ).	<b>K_W09</b>
<b>W_07</b>	Zna i rozumie zagadnienia związane z tworzeniem aplikacji sieciowych bazujących na paradygmacie MVC na platformie .NET.	<b>K_W06, K_W07</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: UMIEJĘTNOŚCI</b>	<b>Symbol efektu kierunkowego</b>
<b>U_01</b>	Potrafi korzystać w efektywny sposób ze środowiska programistycznego Visual Studio 2015 i implementować w nim cztery podstawowe rodzaje projektów (biblioteki klas, aplikacje konsolowe, desktopowe i sieciowe).	<b>K_U10, K_U11</b>
<b>U_02</b>	Potrafi projektować i implementować aplikacje zgodnie z zasadami programowania obiektowego.	<b>K_U10, K_U11</b>
<b>U_03</b>	Potrafi efektywnie korzystać z zaawansowanych mechanizmów i konstrukcji (obsługi wyjątków, interfejsów, generyczności, delegatów, LINQ, refleksji i serializacji).	<b>K_U10, K_U11</b>
<b>U_04</b>	Potrafi ocenić przydatność rutynowych metod i narzędzi służących do rozwiązywania prostych zadań inżynierskich, typowych dla projektowania i programowania aplikacji oraz wybierać i stosować właściwe metody i narzędzia związane z zaawansowanymi technikami programistycznymi.	<b>K_U10, K_U11</b>
<b>U_05</b>	Potrafi zaprojektować, zaimplementować oraz przetestować proste aplikacje sieciowe z dostępem do baz danych i wykorzystaniem wzorca MVC.	<b>K_U19</b>
<b>U_06</b>	Potrafi sformułować i przygotować specyfikację projektowanej aplikacji sieciowej na poziomie realizowanych funkcji.	<b>K_U18</b>

U_07	Potrafi opracować i zaimplementować algorytm w języku C# wykorzystując odpowiednie narzędzia informatyczne.	K_U22
<b>Symbol efektu</b>	<b>Efekt uczenia się: KOMPETENCJE SPOŁECZNE</b>	<b>Symbol efektu kierunkowego</b>
K_01	Jest gotów do podejmowania decyzji i przeprowadzania analizy efektów tych decyzji w ramach rozwiązywania zadań programistycznych.	K_K01
K_02	Jest gotów do uznania znaczenia wiedzy w rozwiązywaniu problemów programistycznych oraz do konstruktywnej krytyki powstałych rozwiązań.	K_K01
<b>Forma i typy zajęć:</b>	studia stacjonarne: wykłady (30 godz.), ćwiczenia laboratoryjne (45 godz.) studia niestacjonarne: wykłady (18 godz.), ćwiczenia laboratoryjne (24 godz.)	
<b>Wymagania wstępne i dodatkowe:</b>		
14. Znajomość podstaw programowania obiektowego. 15. Znajomość podstaw języka C#. 16. Znajomość wzorca MVC.		
<b>Treści modułu kształcenia:</b>		
<ol style="list-style-type: none"> <li><b>Platforma .NET.</b> Omówienie wspólnego silnika uruchomieniowego, wspólnego systemu typów, wspólnej specyfikacji języka, wielojęzyczności i niezależności od platformy.</li> <li><b>Dziedziczenie i polimorfizm.</b> Szczegóły dotyczące filarów programowania obiektowego: klasy nadrzędne i podrzędne, lista inicjalizacyjna, metody wirtualne, nadpisywanie metody, rzutowania jawne i niejawne.</li> <li><b>Strukturalna obsługa wyjątków.</b> Sposoby obsługi sytuacji wyjątkowych w oparciu o system klas dziedziczących po klasie Exception i mechanizm przechwytywania wyjątków.</li> <li><b>Interfejsy.</b> Znaczenie interfejsów jako deklaracji zachowania klasy (w odróżnieniu od jej struktury); definiowanie własnych interfejsów i sposoby korzystania ze standardowych interfejsów.</li> <li><b>Generyczność.</b> Pojęcie typu generycznego, sposoby jego definiowania i wykorzystywania z naciskiem na kolekcje generyczne.</li> <li><b>Delegaty.</b> Wykorzystanie delegatów do komunikacji między obiektami, silnie typowane sposoby przekazywania wskaźników do metod, metody anonimowe, operator lambda.</li> <li><b>Zaawansowane konstrukcje języka C#.</b> Niejawne typowanie zmiennych lokalnych, właściwości automatyczne, metody rozszerzeniowe, metody częściowe, inicjalizator obiektów i typy anonimowe.</li> <li><b>Technologia LINQ.</b> Język zapytań zintegrowany z językiem programowania jako ujednoczony sposób wyszukiwania danych niezależnie od ich źródła. Operatory zapytań i ich wewnętrzna reprezentacja.</li> <li><b>ADO.NET – warstwa połączeniowa.</b> Architektura dostawców danych, wspólne interfejsy i klasy bazowe, korzystanie z obiektów połączenia, polecenia i odczytywania danych.</li> <li><b>ADO.NET – warstwa bezpołączeniowa.</b> Rola warstwy bezpołączeniowej, typy obiektów wykorzystywane do lokalnego reprezentowania bazy danych, sposoby kontroli zmian w lokalnej wersji.</li> </ol>		

11. **LINQ to ADO.NET.** Wykorzystanie technologii LINQ do wyszukiwania danych w bazie danych lub jej lokalnej reprezentacji. Automatyczne generowanie klas reprezentujących tabele i klas wspomagających.
12. Refleksja. Znaczenie metadanych, klasy i metody związane z mechanizmem refleksji; dynamiczne ładowanie pakietów, późne wiązanie, definiowanie własnych atrybutów.
13. **Serializacja.** Znaczenie serializacji, dostępne formaty (binarny, SOAP, XML), konfigurowanie procesu serializacji.
14. **LINQ to XML.** Tworzenie dokumentów XML, ich edytowanie; parsowanie i wyszukiwanie danych w dokumentach XML.
15. **LINQ to Entity.** Entity Framework, podejście Code First, obiekty encji i związane z nimi atrybuty, przykład wykorzystania w ramach aplikacji sieciowej ASP.NET MVC.

#### Literatura podstawowa:

1. Mark Michaelis, Eric Lippert, C# 6.0. Kompletny przewodnik dla praktyków. Wydanie V, Helion 2016
2. Adam Freeman, ASP.NET MVC 5. Zaawansowane programowanie, Helion 2015

#### Literatura dodatkowa:

10. Andrew Troelsen, Philip Japikse, Język C# 6.0 i platforma .NET 4.6, Wydawnictwo Naukowe PWN 2017

#### Planowane formy/działania/metody dydaktyczne:

Wykład tradycyjny wspomagany technikami multimedialnymi, ćwiczenia laboratoryjne wspomagane technikami multimedialnymi. Zamieszczanie na stronach internetowych problemów i zadań ćwiczeniowych.

#### Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:

Efekty **W\_01 – W\_07** będą sprawdzane podczas ćwiczeń i na egzaminie ustnym. Student będzie odpowiadał na pytania dotyczące zaawansowanych pojęć i konstrukcji w programowaniu obiektowym w języku C#:

- Czym jest interfejs? Podaj przykład jego użycia,
- Co to delegaty i do czego ich używamy?
- Na czym polega mechanizm refleksji? Podaj przykłady jego zastosowania.

Studenci zapoznają się z listą wszystkich pytań podczas pierwszego wykładu.

Efekt **U\_01 - U\_07** będą systematycznie sprawdzane na zajęciach. Zadania na następne laboratorium udostępniane są kilka dni wcześniej. Student, na podstawie wykładu mogą się do nich przygotować.

Przykładowe zadania:

- Dana jest lista osób o strukturze z poprzedniego zadania zapisana w liście genrycznej. Napisz odpowiednie zapytania LINQ umożliwiające wyszukanie podanych niżej informacji.
- Zaprojektuj i zaimplementuj interfejsy umożliwiające realizację opisanej poniżej funkcjonalności.
- Zaprojektuj aplikację, którą można będzie rozszerzać za pomocą wtyczek. Opracuj odpowiedni interfejs i mechanizm dynamicznego ładowania pakietów z wtyczkami.



Efekty **K\_01**, **K\_02** będą weryfikowane w oparciu o odpowiedzi na pytania zadawane w czasie ćwiczeń laboratoryjnych oraz podczas zaliczania zadania indywidualnego.

#### **Forma i warunki zaliczenia:**

Przedmiot kończy się egzaminem ustnym. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie ćwiczeń laboratoryjnych. Na zaliczenie ćwiczeń laboratoryjnych składają się oceny częściowe uzyskane na ćwiczeniach z nauczycielem akademickim oraz ocena z samodzielnie wykonanego zadania indywidualnego według schematu:

- Ćwiczenia – 140 punktów (10 punktów za każde z ćwiczeń oprócz ostatniego ćwiczenia przeznaczonego na obrony zadań indywidualnych),
- Zadanie indywidualne – 60 punktów.

Ćwiczenia laboratoryjne będą zaliczone wyłącznie w wypadku uzyskania powyżej połowy punktów z każdego punktowanego ćwiczenia i powyżej połowy punktów z zadania indywidualnego. Na tej formie zajęć student może uzyskać maksymalnie 150 punktów.

Podczas egzaminu ustnego można uzyskać maksymalnie 100 punktów. Egzamin będzie zaliczony w przypadku uzyskania powyżej połowy punktów za każde pytanie. Ocena końcowa z przedmiotu (wystawiana po zaliczeniu wszystkich części składowych), w zależności od sumy uzyskanych punktów (maksymalnie 300 punktów) jest ustalana na podstawie poniższych zakresów (w nawiasach ocena wg skali ECTS):

- 0 – 150 punktów: niedostateczna (F),
- 151 – 180 punktów: dostateczna (E),
- 181 – 210 punktów: dostateczna plus (D),
- 211 – 240 punktów: dobra (C),
- 241 – 2700 punktów: dobra plus (B),
- 271 – 300 punktów: bardzo dobra (A).

#### **Bilans punktów ECTS:**

Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	30 godzin
Udział w ćwiczeniach laboratoryjnych	45 godziny
Udział w konsultacjach z przedmiotu	5 godzin
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	20 godzin
Samodzielne realizacja zadania indywidualnego	15 godzin
Przygotowanie się do egzaminu i obecność na egzaminie	10 godzin

<b>Sumaryczne obciążenie pracą studenta</b>	<b>125 godzin</b>
<b>Punkty ECTS za przedmiot</b>	<b>5 ECTS</b>
Studia niestacjonarne	
Aktywność	Obciążenie studenta
Udział w wykładach	18 godzin
Udział w ćwiczeniach laboratoryjnych	24 godzin
Udział w konsultacjach godz. z przedmiotu	5 godzin
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	43 godzin
Samodzielne realizacja zadania indywidualnego	25 godzin
Przygotowanie się do egzaminu i obecność na egzaminie	10 godzin
<b>Sumaryczne obciążenie pracą studenta</b>	<b>125 godzin</b>
<b>Punkty ECTS za przedmiot</b>	<b>5 ECTS</b>

<b>Sylabus przedmiotu / modułu kształcenia</b>		
<b>Nazwa przedmiotu/modułu kształcenia:</b>	<b>Systemy Baz Danych</b>	
<b>Nazwa w języku angielskim:</b>	Database Systems	
<b>Język wykładowy:</b>	polski	
<b>Kierunek studiów, dla którego przedmiot jest oferowany:</b>	Informatyka	
<b>Jednostka realizująca:</b>	Wydział Nauk Ścisłych i Przyrodniczych	
<b>Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):</b>	fakultatywny	
<b>Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):</b>	pierwszego stopnia	
<b>Rok studiów:</b>	trzeci	
<b>Semestr:</b>	szósty	
<b>Liczba punktów ECTS:</b>	4	
<b>Imię i nazwisko koordynatora przedmiotu:</b>	prof. dr hab. inż. Andrzej Barczak	
<b>Imię i nazwisko prowadzących zajęcia:</b>	prof. dr hab. inż. Andrzej Barczak dr Anna Kołkowicz mgr inż. Monika Berendt-Marchel mgr Zbigniew Młynarski mgr Wojciech Nabiątek	
<b>Założenia i cele przedmiotu:</b>	Celem zajęć jest zapoznanie studentów z zagadnieniami związanymi z systemami baz danych oraz zaprezentowanie najnowszych tendencji ich rozwoju.	
<b>Symbol efektu</b>	<b>Efekt uczenia się: WIEDZA</b>	<b>Symbol efektu kierunkowego</b>
<b>W_01</b>	Zna i rozumie zagadnienia z zakresu teoretycznych podstaw baz danych, systemów baz danych, wykorzystywanych modeli oraz projektowania systemów relacyjnych i obiektowo-relacyjnych baz danych.	<b>K_W09</b>
<b>W_02</b>	Zna aktualny stan oraz najnowsze trendy w rozwoju systemów baz danych relacyjnych i relacyjno-obiektowych.	<b>K_W09</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: UMIEJĘTNOŚCI</b>	<b>Symbol efektu kierunkowego</b>
<b>U_01</b>	Potrafi pozyskiwać informacje na temat systemów baz danych z literatury i innych źródeł, w tym zwłaszcza internetowych; potrafi analizować, interpretować, porządkować, agregować i integrować oraz oceniać pod	<b>K_U01</b>

	względem użyteczności uzyskane informacje, a także wyciągać wnioski na podstawie wyników swojej pracy.	
<b>U_02</b>	Posiada umiejętności samodzielnego pozyskiwania wiedzy w dziedzinie systemów baz danych.	<b>K_U06</b>
<b>U_03</b>	Potrafi ocenić przydatność dostępnych metod i narzędzi, a zwłaszcza środowisk programistycznych służących do rozwiązywania zadań inżynierskich z zakresu systemów baz danych oraz dobierać i stosować właściwe dla rozwiązania określonego problemu metody i narzędzia.	<b>K_U10, K_U11</b>
<b>U_04</b>	Potrafi właściwie oceniać i porównywać systemy baz danych ze względu na wybrane parametry.	<b>K_U15, K_U16, K_U17</b>
<b>U_05</b>	Potrafi, zgodnie ze specyfikacją, zaplanować proces realizacji informatycznego systemu bazodanowego.	<b>K_U11</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: KOMPETENCJE SPOŁECZNE</b>	<b>Symbol efektu kierunkowego</b>
<b>K_01</b>	Jest gotów do podejmowania decyzji, krytycznej oceny działań własnych, działań zespołów, którymi kieruje i organizacji, w których uczestniczy związanych z wykorzystaniem systemów baz i do ponoszenia odpowiedzialności za skutki swoich działań.	<b>K_K01</b>
<b>K_02</b>	Jest gotów do uznania znaczenia wiedzy w rozwiązywaniu problemów związanych z systemami baz danych oraz jest gotów do konstruktywnej krytyki w stosunku do działań swoich i innych osób.	<b>K_K01</b>
<b>Forma i typy zajęć:</b>	Studia stacjonarne: wykłady (30 godz.), ćwiczenia laboratoryjne (45 godz.) Studia niestacjonarne: wykłady (18 godz.), ćwiczenia laboratoryjne (24 godz.)	
<b>Wymagania wstępne i dodatkowe:</b>		
Wiedza z zakresu baz danych, systemów operacyjnych oraz inżynierii oprogramowania		
<b>Treści modułu kształcenia:</b>		
<ol style="list-style-type: none"> <li><b>Język PL/SQL (cz.1).</b> Wstęp do programowania w PL/SQL. Typy danych. Tworzenie bloków PL/SQL.</li> <li><b>Język PL/SQL (cz. 2).</b> Wyrażenia, operatory, funkcje i wyjątki. Struktury sterujące.</li> <li><b>Język PL/SQL (cz. 3).</b> Polecenia SQL w programie PL/SQL. Wyzwalacze. Przechowywanie procedur i funkcji. Pakiety. Kursory. Przegląd funkcji wbudowanych.</li> <li><b>Język T-SQL</b> Omówienie różnic między T-SQL a PL/SQL. Microsoft Common Table Expressions, rekursja w zapytaniach, konwersja wyników do XML/JSON, rankingi.</li> <li><b>System zarządzania bazą danych.</b> Zarządzanie plikami, wyszukiwanie informacji. Jądro SZBD, model pamięci zewnętrznej. Pliki nieuporządkowane, pliki sekwencyjne. Pliki haszowane, pliki indeksowe, indeks a struktura B-drzewa.</li> <li><b>System zarządzania bazą danych (cz. 1).</b> Zarządzanie transakcjami. Współbieżność transakcji - sytuacje konfliktowe. Metody zapewniające współbieżność transakcji. Metody blokowania. Zarządzanie integralnością bazy danych.</li> </ol>		

7. **System zarządzania bazą danych (cz. 2).** Niezawodność i odtwarzanie baz danych po awarii. Obrazy przed i po transakcji, uaktualnianie segmentów pamięci. Tabele słownikowe. Zarządzanie zapytaniami, optymalizacja zapytań.
8. **Inteligentne bazy danych (cz. 1).** Inteligencja, a bazy danych. Logika i bazy danych. Definicja danych w katalogu, operowanie danymi w Datalogu. Obsługa DDL w PL/SQL i T-SQL.
9. **Inteligentne bazy danych (cz. 2).** Datalog i relacyjny model danych. Różnice między Datalogiem i Prologiem. Systemy hipermedialne, geograficzne systemy informacyjne. Zasady budowy i działania aplikacji pracującej w architekturze trójwarstwowej.
10. **Architektura bazy danych ORACLE (cz. 1).** Baza danych i instancje. Wewnętrzna struktura bazy danych. Wewnętrzne obszary pamięci. Zasady budowy formularzy i przetwarzania wprowadzonych danych.
11. **Architektura bazy danych ORACLE (cz. 2).** Procesy drugoplanowe. Podstawowa konfiguracja bazy danych. Model danych i tworzenie bazy danych. Autoryzacja w aplikacji bazodanowej.
12. **Architektura bazy danych MS SQL Server.** Baza danych i instancje. Wewnętrzna struktura bazy danych. Wewnętrzne obszary pamięci. Podstawowa konfiguracja bazy danych. Model danych i tworzenie bazy danych. Autoryzacja w aplikacji bazodanowej.
13. **Konfiguracja sprzętowa baz danych ORACLE i MS SQL Server (cz. 1).** Połączone bazy danych. Zdalna modyfikacja danych. Zalety rozproszenia.
14. **Konfiguracja sprzętowa baz danych ORACLE i MS SQL Server (cz. 2).** Serwery klastrowe, konfiguracje wieloprocessorowe. Aplikacje typu klient-serwer. Architektura trójwarstwowa. Operacje sieciowe z użyciem wbudowanych pakietów.
15. **Porównanie systemów bazodanowych.** Różnice i podobieństwa w tworzeniu systemów bazodanowych m. in. w ORACLE, Microsoft SQL Server, PostgreSQL, MariaDB. Wybór odpowiedniego narzędzia do rozwiązywania określonych problemów bazodanowych.

#### Literatura podstawowa:

1. Connolly T., Begg C.: Systemy baz danych - Praktyczne metody projektowania, implementacji i zarządzania. Tom 1, 2; Wydawnictwo RM, 2004
2. Darwen H., Date C.J.: SQL. Omówienie standardu języka; Wydawnictwa Naukowo Techniczne, 2000
3. Elmasri R., Navathe S. B.: Wprowadzenie do systemów baz danych; Wydawnictwo Helion, 2005
4. Allen S.: Modelowanie danych; Wydawnictwo Helion, 2006C. J. Date , Wprowadzenie do systemów baz danych, WNT, 2000
5. Garcia-Molina H., Ullman J. D., Widom J.; Systemy baz danych Pełny wykład; Wydawnictwa Naukowo Techniczne, 2006
6. Dokumentacja Oracle: Application Developer's Guide - Fundamentals, PL/SQL User's Guide and Reference, PL/SQL Web Toolkit Reference, Using the PL/SQL Gateway.

#### Literatura dodatkowa:

1. Andrzej Barczak, Dariusz Zacharczuk, Stanisław Jastrzebowski, Comparative analysis of database access technology w Studia Informatica. System and information technology, Wyd. AP, Siedlce 2009 (vol. 12)
2. Andrzej Barczak, Dariusz Zacharczuk, Damian Pluta, Tools and methods of databases optimization In Oracle 10g. Part 1. Tuning instance. w Studia Informatica.1-2 (16) 2012, Wyd.UPH, Siedlce
3. Andrzej Barczak, Dariusz Zacharczuk, Anna Korzeniecka, The influence of indexing methods on effective functioning of the database w Studia Informatica.1-2 (17) 2013, Wyd.UPH, Siedlce
4. Andrzej Barczak, Dariusz Zacharczuk, Damian Pluta, Tools and methods of databases optimization In Oracle 10g. Part 2. Tuning of hardware applications and SQL queries w Studia Informatica.1 (18) 2014, Wyd.UPH, Siedlce
5. Andrzej Barczak, Dariusz Zacharczuk, Damian Pluta, Tools and methods of databases optimization In Oracle 10g. Part 3. Theory in practice. w Studia Informatica.1 (18) 2014 , Wyd.UPH, Siedlce

**Planowane formy/działania/metody dydaktyczne:**

Wykład tradycyjny wspomagany technikami multimedialnymi, laboratoria – praktyczna praca na komputerze. Zamieszczanie na stronach internetowych zagadnień teoretycznych i zadań ćwiczeniowych

**Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:**

Efekty W\_01 i W\_02 będą weryfikowane na egzaminie pisemnym.

Efekty U\_01 - U\_05 weryfikowane będą w trakcie zajęć laboratoryjnych.

Efekty K\_01 - K\_02 będą weryfikowane, w oparciu o posiadaną wiedzę i umiejętności, w czasie zajęć laboratoryjnych. W trakcie zaliczania zadania indywidualnego, a także będą sprawdzane na egzaminie.

Przykładowe pytania egzaminacyjne:

- Omów funkcje jądra SZBD,
- Omów wewnętrzne obiekty zaawansowanej bazy danych (ORACLE, SQL Serwer),
- Przedstaw architekturę zaawansowanej bazy danych (ORACLE, SQL Serwer),
- Zdefiniuj wyzwalacz aktualizujący dane zawarte w zaawansowanej bazie danych,
- Obsłuż sytuacje wyjątkowe zaistniałe przy przetwarzaniu danych w bazie danych.

Przed egzaminem studenci będą mieli dostęp do pełnej listy pytań.

**Forma i warunki zaliczenia:**

Moduł kończy się egzaminem. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie laboratorium. Na zaliczenie laboratorium składają się średnie ważone punktów uzyskanych na regularnych laboratoriach (60%) oraz punkty z wykonanego zadania indywidualnego (40%).

Zajęcia laboratoryjne będą zaliczone w wypadku uzyskania:

- minimum 51% z regularnych zajęć (61/120 pkt)
- minimum 51% z zadania indywidualnego (16/30 pkt).

Maksymalne z tej części zajęć można uzyskać 150 pkt.

Egzamin jest egzaminem pisemnym. Można na nim uzyskać maksymalnie 50 pkt. Egzamin będzie zaliczony w przypadku uzyskania co najmniej 25 pkt. Ocena końcowa z modułu (wystawiana po zaliczeniu wszystkich części składowych), w zależności od sumy uzyskanych punktów (maksymalnie 200 pkt) jest następująca (w nawiasach ocena w skali ECTS):

- 0 – 100 - niedostateczna (2,0) (F)
- 101 - 120 - dostateczna (3,0) (E)
- 121 - 140 - dostateczna plus (3,5) (D)
- 141 - 160 - dobra (4,0) (C)
- 161 - 180 - dobra plus (4,5) (B)
- 181 - 200 - bardzo dobra (5,0).(A)

Poprawy: jednorazowa poprawa egzaminu pisemnego. Jednorazowe poprawy do 2 ćwiczeń laboratoryjnych w trakcie trwania semestru.

**Bilans punktów ECTS:**

Studia stacjonarne

Aktywność

Obciążenie studenta

Udział w wykładach

30 godz.

Udział w ćwiczeniach	45 godz.
Udział w konsultacjach z przedmiotu	5 godz.
Samodzielne przygotowanie się do ćwiczeń	10 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	10 godz.
Sumaryczne obciążenie pracą studenta	<b>100 godz.</b>
Punkty ECTS za przedmiot	4 ECTS
Studia niestacjonarne	
Aktywność	Obciążenie studenta
Udział w wykładach	18 godz.
Udział w ćwiczeniach	24 godz.
Udział w konsultacjach z przedmiotu	3 godz.
Samodzielne przygotowanie się do ćwiczeń	35 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	20 godz.
Sumaryczne obciążenie pracą studenta	<b>100 godz.</b>
Punkty ECTS za przedmiot	4 ECTS

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:		Aplikacje internetowe i rozproszone
Nazwa w języku angielskim:		Internet and distributed applications
Język wykładowy:	polski	
Kierunek studiów, dla którego przedmiot jest oferowany:		informatyka
Jednostka realizująca:	Wydział Nauk Ścisłych i Przyrodniczych	
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):		fakultatywny
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):		pierwszego stopnia
Rok studiów:	trzeci	
Semestr:	Szósty	
Liczba punktów ECTS:	4	
Imię i nazwisko koordynatora przedmiotu:		Dr Dariusz Mikułowski
Imię i nazwisko prowadzących ćwiczenia:		Dr Dariusz Mikułowski
Założenia i cele przedmiotu:		Celem kursu jest zapoznanie studentów z projektowaniem i implementacją aplikacji opartych o zaawansowane techniki i technologie rozproszone i internetowe
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	Student zna i rozumie funkcjonowanie języków opisu dokumentu HTML i XML i ich zastosowania.	K_W12
W_02	Zna i rozumie zasady działania języków opisu struktury dokumentu DTD oraz XMLSchema	K_W12
W_03	Zna i rozumie zasadę działania języków programowania po stronie klienta JavaScript	K_W12
W_04	Zna i rozumie działanie języki zapytań do dokumentów XML: XPath oraz XQuery.	K_W12
W_05	Zna i rozumie koncepcje architektur otwartych systemów internetowych: WebService, WS-REST oraz P2P. oraz ich zastosowania.	K_W12
Symbol efektu	Efekt uczenia się: UMIEJĘTNOŚCI	Symbol efektu kierunkowego
U_01	Potrafi pisać programy posługujące się pocztą internetową jako medium komunikacji	K_U09, K_U10
U_02	Potrafi wykorzystać w komunikacji między aplikacjami język XML oraz HTML i specyfikować język komunikacji w DTD oraz XMLSchema.	K_U09, K_U10



U_03	Potrafi pisać programy po stronie klienta w języku JavaScript, w tym rysować grafikę	K_U09, K_U10
U_04	Posługuje się narzędziami do wyszukiwania informacji w dokumentach XML/HTML oraz do ich przetwarzania.	K_U09, K_U10
Symbol efektu	<b>Efekt uczenia się: KOMPETENCJE SPOŁECZNE</b>	<b>Symbol efektu kierunkowego</b>
K_01	Pojmuje naukę jako postępowy rozwój teoretycznych uogólnień wynikający z obserwacji i doświadczeń i prowadzący do nowych obserwacji i nowych doświadczeń.	K_K01
K_02	rozumie jedność teorii i praktyki oraz konieczność pogłębiania swej wiedzy w sposób ukierunkowany, by stawiać czoła konkretnym wyzwaniom projektowym	K_K01, K_K03
K_03	Potrafi oceniać wartość informacji, szczególnie dostępnej w Internecie, nie tylko na poziomie statystycznym, ale i syntaktycznym, semantycznym, pragmatycznym i apobetycznym.	K_K03
<b>Forma i typy zajęć:</b>	studia stacjonarne: wykłady (30 godz.), ćwiczenia laboratoryjne (45 godz.) studia niestacjonarne: wykłady (18 godz.), ćwiczenia laboratoryjne (24 godz.)	
<b>Wymagania wstępne i dodatkowe:</b>		
Umiejętność programowania		
<b>Treści modułu kształcenia:</b>		
<ol style="list-style-type: none"> <li><b>Infrastruktura i architektury dla aplikacji internetowych:</b> model wielowarstwowy ISO-OSI sieci komputerowych, wybrane protokoły internetowe, poczta elektroniczna</li> <li><b>WWW jako przykład architektury klient-serwer:</b> HTTP, wybrane serwery WWW</li> <li><b>język HTML:</b> struktura języka i jego przeznaczenie, wsparcie dla dynamicznego HTML, struktura CSS i jego przeznaczenie, architektura DOM dokumentów internetowych</li> <li><b>Język XML (eXtensible Markup Language) jako język wymiany informacji między aplikacjami:</b> XML jako szczególny przypadek SGML, budowa dokumentu, przestrzenie nazw, zastosowania XML</li> <li><b>JavaScript:</b> struktura języka i jego przeznaczenie, wsparcie dla dynamicznego HTML (struktury DOM)</li> <li><b>Język XML (eXtensible Markup Language) - interfejsy programistyczne:</b> model DOM, model SAX, realizacje w Javie</li> <li><b>Metainformacje dla XML - słaba kontrola składni:</b> DTD, XSchema</li> <li><b>Język ścieżek XPATH oraz język transformacji dokumentów XML (XSLT):</b> XPath, koncepcja transformacji XSL (XML Stylesheet Language);, Budowa wzorców XSLT; , Środki programistyczne dostępne w XSLT.</li> <li><b>repozytoria XML i języki zapytań:</b> repozytoria XML, XQUERY, inne języki zapytań dla XML</li> <li><b>Problemy technologiczne wyszukiwarek internetowych:</b> Budowa wyszukiwarki, Technologie pajaków Technologie Indeksów, Metody wyszukiwania</li> <li><b>Java beans:</b> struktura EJB, zastosowania</li> <li><b>Otwarte architektury aplikacji: SOAP i Web-Serwisy:</b> Oprogramowanie pośredniczące, WSDL, ebXML</li> <li><b>Otwarte architektury aplikacji: REST:</b> Wady koncepcji Webservisów, Naturalność architektury REST</li> <li><b>Otwarte architektury aplikacji: Modele "koleżeńskie":</b> architektura P2P, wymagania technologiczne, JXTA, zastosowania w biznesie</li> <li><b>Mobilni agenci:</b> koncepcja agenta, mobilność, platformy, zastosowania</li> </ol>		

16. Programowanie w chmurze: środki programistyczne, kontrola zasobów, aspekty ekonomiczne

**Literatura podstawowa:**

1. M. A. Kłopotek, S.T. Wierzchoń, K. Ciesielski, Michał Dрамиński, D. Czerski: Conceptual Maps of Document Collections in Internet and Intranet. Coping with Technological Challenge. IPI PAN Publishing House, 2007. 139p
2. Java Servlet i JavaServer Pages. Tom 1. Wydanie II Autorzy: Marty Hall, Larry Brown Tłumaczenie: Piotr Rajca ISBN: 83-246-0032-9 Tytuł oryginału: Core Servlets and JavaServer Pages, Vol. 1: Core Technologies, Second Edition Data wydania: 11/2005

**Literatura dodatkowa:**

1. B. Eckel: Thinking in Java Prentice-Hall, December 2002
2. Dokumentacja Javy jdk 1.7 i innych pakietów Javy
3. M.A.Kłopotek: Inteligentne wyszukiwarki internetowe. Akademicka Oficyna Wydawnicza Exit, Warszawa 2001, 332 strony, ISBN 83-87674-31-1
4. Java 1.5 Tiger. Zapiski programisty Autorzy: Brett McLaughlin, David Flanagan Tłumaczenie: Jaromir Senczyk ISBN: 83-246-0048-5 Tytuł oryginału: Java 1.5 Tiger A Developers Notebook Format: B5, stron: 224 Data wydania: 01/2006
5. XSLT dla każdego Autor: Michiel van Otegem Tłumaczenie: Tomasz Żmijewski ISBN: 83-7197-785-9 Tytuł oryginału: TY XSLT in 21 Days Format: B5, stron: 576 Data wydania: 05/2003
6. Head First Servlets & JSP. Edycja polska Autorzy: Bryan Basham, Kathy Sierra, Bert Bates Tłumaczenie: Piotr Rajca, Mikołaj Szczepaniak ISBN: 83-7361-810-4 Tytuł oryginału: Head First Servlets & JSP Data wydania: 07/2005
7. Dokumentacja pakietu atomiki i polskieznaki (zawarta w Warcabkach.java

**Planowane formy/działania/metody dydaktyczne:**

Wykład tradycyjny wspomagany technikami multimedialnymi, ćwiczenia rachunkowe wspomagane technikami multimedialnymi. Zamieszczanie na stronach internetowych problemów i zadań ćwiczeniowych.

**Sposoby weryfikacji efektów kształcenia osiągniętych przez studenta:**

Podczas ćwiczeń laboratoryjnych sprawdzane będą efekty U\_01 - U\_04. Podczas egzaminu efekty U\_01 - U\_04 oraz W\_01-W\_05.

**Forma i warunki zaliczenia:**

Moduł kończy się egzaminem. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie laboratorium. Na zaliczenie laboratorium składają się oceny cząstkowe uzyskane na regularnych zajęciach z nauczycielem akademickim, za które można uzyskać maksymalnie 40 pkt. Zaliczenie zajęć laboratoryjnych i dopuszczenie do egzaminu jest możliwe po uzyskaniu co najmniej 20 pkt.

Egzamin jest egzaminem pisemnym. Można na nim uzyskać do 60 pkt. Egzamin będzie zaliczony w przypadku uzyskania co najmniej 30 pkt. Ocena końcowa z modułu (wystawiana po zaliczeniu wszystkich części składowych), w 0 – 50 pkt: niedostateczna (F),

- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).Poprawy:

Dwukrotna poprawa egzaminu w sesji. Jednorazowe poprawy do 5 ćwiczeń laboratoryjnych w trakcie semestru.

Bilans punktów ECTS:

Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	30 godz.
Udział w ćwiczeniach laboratoryjnych	45 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	10 godz.
Udział w konsultacjach godz. z przedmiotu	5 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	10 godz.
Sumaryczne obciążenie pracą studenta	100 godz.
Punkty ECTS za przedmiot	4 ECTS

Studia niestacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	18 godz.
Udział w ćwiczeniach laboratoryjnych	24 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	35 godz.

Udział w konsultacjach godz. z przedmiotu	3 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	20 godz.
Sumaryczne obciążenie pracą studenta	100 godz.
Punkty ECTS za przedmiot	4 ECTS

<b>Sylabus przedmiotu / modułu kształcenia</b>		
<b>Nazwa przedmiotu/modułu kształcenia:</b>	<b>Bezpieczeństwo systemów internetowych i baz danych</b>	
<b>Nazwa w języku angielskim:</b>	Security of internet systems and databases	
<b>Język wykładowy:</b>	polski	
<b>Kierunek studiów, dla którego przedmiot jest oferowany:</b>	informatyka	
<b>Jednostka realizująca:</b>	Wydział Nauk Ścisłych i Przyrodniczych	
<b>Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):</b>	fakultatywny	
<b>Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):</b>	pierwszego stopnia	
<b>Rok studiów:</b>	trzeci	
<b>Semestr:</b>	szósty	
<b>Liczba punktów ECTS:</b>	3	
<b>Imię i nazwisko koordynatora przedmiotu:</b>	dr Piotr Świtalski	
<b>Imię i nazwisko prowadzących zajęcia:</b>	dr Piotr Świtalski	
<b>Założenia i cele przedmiotu:</b>	<p>Celem przedmiotu jest zaznajomienie studentów z problematyką bezpieczeństwa systemów komputerowych. Przedmiot ma za zadanie zapoznanie z klasycznymi technikami bezpieczeństwa w systemach komputerowych. W założeniach do tego przedmiotu przewiduje się zajęcia praktyczne z użyciem komputerów, podczas których studenci nabędą umiejętności z atakami na systemy komputerowe i również metody przeciwdziałania im. Dodatkowo studenci poznają techniki zabezpieczania baz danych. Ma również usystematyzować wiedzę w zakresie standardów bezpieczeństwa.</p>	
<b>Symbol efektu</b>	<b>Efekt uczenia się: WIEDZA</b>	<b>Symbol efektu kierunkowego</b>
<b>W_01</b>	Zna i rozumie zagadnienia z zakresu bezpieczeństwa systemów komputerowych	<b>K_W05, K_W07</b>

<b>W_02</b>	Zna i rozumie standardy i uwarunkowania prawne związane z bezpieczeństwem systemów komputerowych	<b>K_W04, K_W10</b>
<b>W_03</b>	Zna i rozumie techniki związane z atakami na systemy komputerowe w tym na bazy danych i potrafi im przeciwdziałać	<b>K_W05, K_W07</b>
<b>W_04</b>	Zna i rozumie mechanizmy bezpieczeństwa komputerowego i potrafi je wykorzystać w przedstawionym środowisku	<b>K_W07, K_W12, K_W15</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: UMIEJĘTNOŚCI</b>	<b>Symbol efektu kierunkowego</b>
<b>U_01</b>	Potrafi sprawnie wyszukiwać w literaturze informacje związane z bezpieczeństwem systemów komputerowych, potrafi wyszukać informacje na temat nowych podatności wykrytych w systemach komputerowych	<b>K_U01</b>
<b>U_02</b>	Potrafi sprawnie wykorzystać narzędzia bezpieczeństwa systemów komputerowych	<b>K_U02, K_U17</b>
<b>U_03</b>	Potrafi dobrać adekwatne zabezpieczenia w stosunku do zagrożenia w systemie komputerowym	<b>K_U10, K_U17, K_U25</b>
<b>Forma i typy zajęć:</b>	studia stacjonarne: wykłady (30 godz.), ćwiczenia laboratoryjne (30 godz.) studia niestacjonarne: wykłady (15 godz.), ćwiczenia laboratoryjne (15 godz.)	
<b>Wymagania wstępne i dodatkowe:</b>		
Warunkiem uczestnictwa w zajęciach jest uprzednie zaliczenie następujących przedmiotów: „Architektura systemów komputerowych”, „Systemy operacyjne”, „Technologie sieciowe”, „Bazy danych” lub znajomość literatury obowiązującej w tych przedmiotach.		
<b>Treści modułu kształcenia:</b>		
<ol style="list-style-type: none"> <li><b>Koncepcja bezpieczeństwa komputerowego.</b> Podstawowe zasady. Właściwości i klasyfikacja koncepcji bezpieczeństwa: integralność, dostępność, poufność, niezaprzeczalność, odpowiedzialność. Podstawowe pojęcia. Model bezpieczeństwa sieciowego. Minimalne standardy bezpieczeństwa. Polityka bezpieczeństwa.</li> <li><b>Przestępczość komputerowa.</b> Przesłęstwa ujawnione i nieujawnione. Obiekty, typy i sprawcy przestępcstw. Piractwo komputerowe, sabotaż, wywiad gospodarczy, szpiegostwo, przestępcstwa bankowe. Inżynieria społeczna. Phising. Inżynieria odwrotna. Organizacje przeciwdziałające przestępczości komputerowej.</li> <li><b>Aspekty prawne ochrony informacji i normalizacja.</b> Stan prawny ochrony informacji w Polsce. Ochrona danych osobowych. Hacking. Prawo autorskie a programy komputerowe. Normalizacja w zakresie bezpieczeństwa systemów informatycznych.</li> </ol>		

4. **Charakterystyka zagrożeń w systemach informatycznych.** Specyfika systemów informatycznych. Klasyfikacja zagrożeń. Zagrożenia związane z DNS. Złośliwe oprogramowanie. Sieci botnet. Statystyka typowych zagrożeń.
5. **Klasyczne techniki szyfrowania.** Dziedzina kryptografii i podstawowe pojęcia. Podstawowe techniki szyfrowania: technika podstawieniowa, szyfr Cezara, szyfry mono i polialfabetyczne, szyfr Playfaira, szyfr Vigenère'a. Techniki przestawieniowe, szyfr zygzakowy, maszyny wirnikowe. Szyfrowanie symetryczne. Ataki siłowe przeprowadzane na algorytmy szyfrowania.
6. **Szyfry strumieniowe i blokowe.** Szyfry strumieniowe. Szyfr strumieniowy RC4. Szyfry blokowe. Struktura i szyfr Feistela. Standard DES, efekt lawiny. Algorytm AES. Tryby operacyjne szyfrów blokowych. Szyfrowanie asymetryczne. Algorytm RSA.
7. **Podpis cyfrowy.** Idea podpisu cyfrowego. Wymagania stawiane podpisom cyfrowym. Mechanizm uwierzytelniania komunikatów. Kody uwierzytelnienia komunikatów MAC. Podpis cyfrowy ElGamal. Standard DSS. Algorytm DSA. Kryptograficzne funkcje haszujące. Algorytm SHA-512. Paradoks urodzin.
8. **Uwierzytelnianie.** Pojęcie uwierzytelniania. Uwierzytelnianie przez hasło. Strategie wyboru haseł. Inne metody uwierzytelniania. Protokoły uwierzytelniania: protokół challenge and response. Atak „człowiek pośrodku”. Dowód z wiedzą zerową. Uwierzytelnianie dwuskładnikowe. Hasło jednorazowe. Generowanie haseł jednorazowych – protokół S/KEY.
9. **Szkodliwe oprogramowanie.** Problemy związane z nieautoryzowanym dostępem do systemów komputerowych. Infekcje systemów komputerowych. Bezpieczny system operacyjny. Podatność systemów operacyjnych. Charakterystyka szkodliwego oprogramowania: tylne drzwi, bomby logiczne, konie trojańskie, wirusy, robaki, exploity i rootkity, keylogery. Przeciwdziałanie szkodliwemu oprogramowaniu.
10. **Nadużycia w sieciach rozległych.** Protokoły komunikacyjne. Struktura nagłówek protokołów TCP/IP. Ataki na warstwę sieciową. Fałszowanie pakietów IP. Fragmentacja pakietów IP. Ataki na warstwę transportową. Atak SYN Flood. Skanowanie portów. Ataki w sieciach komórkowych.
11. **Zapory sieciowe (firewalle).** Model ogólny zapory sieciowej. Charakterystyka firewalli. Ograniczenia firewalli. Firewall filtrujący pakiety. Firewall filtrujący pakiety z badaniem stanu pakietu. Brama aplikacyjna, brama transmisyjna. Implementacja firewalla. Strefa zdemilitaryzowana (DMZ). Przykładowa konfiguracja firewalla z DMZ.
12. **Systemy wykrywania intruzów.** Zachowania intruzywne. Wzorce zachowań intruzów. Wykrywanie intruzów. Statystyczna analiza zachowania. Wykrywanie intruzów w oparciu o reguły. Systemy IDPS. Audyt w systemach IDPS. Pułapki (Honeypoty).
13. **Bezpieczeństwo systemów internetowych.** Ataki w warstwie aplikacji. Przepelnienie bufora. Wstrzykiwanie kodu. Ataki SQL Injection. Błędy w uwierzytelnianiu użytkownika i zarządzania jego sesją. Atak XSS. Błędy w konfiguracji oprogramowania. Niewłaściwe zabezpieczenie wrażliwych danych. Atak Cross-Site Request Forgery (CSRF).
14. **Bezpieczeństwo baz danych.** Bezpieczeństwo serwera bazy danych. Bezpieczeństwo fizyczne danych. Poufność danych w bazach danych. Mechanizmy uwierzytelniania w systemach baz danych. Zarządzanie użytkownikami w bazach danych i konfiguracja dostępu do danych. Szyfrowanie danych. Monitorowanie aktywności w bazach danych. Bezpieczeństwo aplikacji wykorzystujących bazy danych.
15. **Zapewnianie dostępności danych.** Utrzymanie ciągłości zasilania. Sposoby zapobiegania problemom zasilania, zasilacze awaryjne UPS. Ochrona danych przed utratą. Systemy macierzowe RAID. Kopie bezpieczeństwa.

### Literatura podstawowa:

1. Stallings W.: Kryptografia i bezpieczeństwo sieci komputerowych. Matematyka szyfrów i techniki kryptologii, Wyd. Helion, Gliwice, 2012.
2. Prasad P.: Testy penetracyjne nowoczesnych serwisów. Kompendium inżynierów bezpieczeństwa, Wyd. Helion, Gliwice, 2017.
3. Stuttard D., Pinto M., The Web Application Hacker's Handbook, John Wiley & Sons, Inc, 2011.
4. Barczak A., Sydoruk T.: Bezpieczeństwo systemów informatycznych. Wyd. Akademii Podlaskiej, Siedlce, 2002.

### Literatura dodatkowa:

1. Stallings W.: Kryptografia i bezpieczeństwo sieci komputerowych. Konceptcje i metody bezpiecznej komunikacji, Wyd. Helion, Gliwice, 2012.
2. Schetina E., Green K., Carlson J.: Bezpieczeństwo w sieci, Wyd. Helion, Gliwice, 2002.
3. Suehring S.: Zapory sieciowe w systemie Linux. Kompendium wiedzy o nftables. Wydanie IV, Wyd. Helion, Gliwice, 2015.

### Planowane formy/działania/metody dydaktyczne:

Wykład tradycyjny wspomagany jest technikami multimedialnymi. Ćwiczenia laboratoryjne – zajęcia praktyczne z wykorzystaniem wybranych narzędzi programowych. Na stronie internetowej prowadzącego zamieszczone są materiały z problemami i zadaniami laboratoryjnymi.

### Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:

Efekty W\_01 do W\_04 weryfikowane będą poprzez egzamin pisemny, a także w toku weryfikacji przygotowania do kolejnych zajęć laboratoryjnych. Na egzaminie pisemnym pytania będą dotyczyły poznanych technik ataków i sposobów zabezpieczania systemów komputerowych. Egzamin będzie również obejmował treści związane z kryptografią. Przykładowe pytania:

- Przedstaw schemat działania podpisu cyfrowego.
- Na czym polega uwierzytelnianie dwuskładnikowe?
- W jaki sposób przeprowadzany jest atak DDoS?
- Na czym polega szyfrowanie symetryczne?
- Omów metody zabezpieczania baz danych.

Przed egzaminem studenci będą mieli dostęp do przykładowych pytań na egzamin.

Efekty U\_01 do U\_03 będą sprawdzane systematycznie na zajęciach laboratoryjnych. Przykładowe zadania:

- Zaimplementuj certyfikat SSL i zaimplementuj go w wybranej przeglądarce internetowej.
- Napisz polecenie dla zapory iptables, które będzie odrzucało wszystkie nowe połączenia TCP z adresu IP o adresie 91.205.12.6 na porcie 22.

Materiały na następne laboratorium będą dostępne na dwa dni przed zajęciami.

### Forma i warunki zaliczenia:



Ocena z przedmiotu składa się z dwóch ocen cząstkowych:

- oceny z zajęć laboratoryjnych,
- oceny z egzaminu końcowego.

Na ocenę z zajęć laboratoryjnych składają się oceny cząstkowe uzyskane na regularnych zajęciach z nauczycielem akademickim, za które można uzyskać sumarycznie 40 pkt. Zaliczenie zajęć laboratoryjnych możliwe po uzyskaniu co najmniej 51% liczby punktów z tej formy zaliczenia

Egzamin jest egzaminem pisemnym. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie laboratorium. Można na nim uzyskać maksymalnie 60 pkt. Egzamin będzie zaliczony w przypadku uzyskania co najmniej 51% liczby punktów z tej formy zaliczenia. Ocena końcowa z przedmiotu, w zależności od sumy uzyskanych punktów (maksymalnie 100 pkt) jest następująca (w nawiasach ocena wg skali ECTS):

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

#### **Bilans punktów ECTS:**

##### Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	30 godz.
Udział w ćwiczeniach laboratoryjnych	30 godz.
Przygotowanie się do egzaminu	8 godz.
Przygotowanie się do ćwiczeń laboratoryjnych	5 godz.
Udział w konsultacjach z przedmiotu	2 godz.
<b>Sumaryczne obciążenie pracą studenta</b>	<b>75 godz.</b>
<b>Punkty ECTS za przedmiot</b>	<b>3 ECTS</b>

##### Studia niestacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	15 godz.
Udział w ćwiczeniach laboratoryjnych	15 godz.

Przygotowanie się do egzaminu	20 godz.
Przygotowanie się do ćwiczeń laboratoryjnych	20 godz.
Udział w konsultacjach z przedmiotu	5 godz.
<b>Sumaryczne obciążenie pracą studenta</b>	<b>75 godz.</b>
<b>Punkty ECTS za przedmiot</b>	<b>3 ECTS</b>

**Specjalność: Systemy i sieci komputerowe, semestr VI**

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:	Zaawansowane sieci komputerowe	
Nazwa w języku angielskim:	Advanced Computer Networking	
Język wykładowy:	polski	
Kierunek studiów, dla którego przedmiot jest oferowany:	informatyka	
Jednostka realizująca:	Wydział Nauk Ścisłych i Przyrodniczych	
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):	fakultatywny	
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):	pierwszego stopnia	
Rok studiów:	trzeci	
Semestr:	szósty	
Liczba punktów ECTS:	5	
Imię i nazwisko koordynatora przedmiotu:	Prof. dr hab. Stanisław Ambroszkiewicz	
Imię i nazwisko prowadzących zajęcia:	dr Andrzej Salamończyk dr Grzegorz Terlikowski	
Założenia i cele przedmiotu:	Celem modułu jest zaznajomienie studentów z zaawansowanymi protokołami i technologiami sieci komputerowych	
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	Student zna i rozumie zasadę działania sieciowych protokołów komunikacyjnych	K_W07
W_02	Zna i rozumie zasadę przełączania w sieciach LAN (VLANy).	K_W07
W_03	Zna i rozumie zasadę projektowania sieci WAN, PPP, ISDN, Frame Relay.	K_W07
W_04	Zna i rozumie wybrane aspekty bezpieczeństwa sieci komputerowych.	K_W07
W_05	Zna i rozumie wybrane elementy programowania sieciowego w oparciu o język Java.	K_W07
Symbol efektu	Efekt uczenia się: UMIEJĘTNOŚCI	Symbol efektu kierunkowego
U_01	Potrafi realizować aplikacje sieciowe	K_U10, K_U11, K_U12, K_U19
U_02	Potrafi zaprojektować, zrealizować i skonfigurować sieć LAN, VLAN i WAN	K_U10, K_U21
U_03	Potrafi diagnozować i usuwać problemy w sieciach LAN, VLAN i WAN	K_U10, K_U21

<b>U_04</b>	Potrafi zaprojektować i skonfigurować podstawą politykę bezpieczeństwa w sieciach z wykorzystaniem ACL	<b>K_U10, K_U23</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: KOMPETENCJE SPOŁECZNE</b>	<b>Symbol efektu kierunkowego</b>
<b>K_01</b>	Jest gotów do podejmowania decyzji i krytycznej oceny własnych rozwiązań w rozwiązywaniu problemów w istniejących sieciach komputerowych uwzględniając istniejące standardy sieciowe	<b>K_K01</b>
<b>Forma i typy zajęć:</b>	studia stacjonarne: wykłady (30 godz.), ćwiczenia laboratoryjne (45 godz.) studia niestacjonarne: wykłady (18 godz.), ćwiczenia laboratoryjne (24 godz.)	
<b>Wymagania wstępne i dodatkowe:</b>		
1. Umiejętność programowania w języku obiektowym (Java). 2. Znajomość podstawowych pojęć technologii sieciowych (zakres przedmiotu Technologie Sieciowe).		
<b>Treści modułu kształcenia:</b>		
Treści wykładu: <ol style="list-style-type: none"> <li>1. <b>Warstwa aplikacji - programowanie w oparciu o gniazda. Gniazda (sockety) UDP. Zastosowania protokołu UDP. Programowanie na socketach UDP.</b></li> <li>2. <b>Warstwa aplikacji - programowanie w oparciu o URL. Protokół HTTP, adresacja URL, HTML. Model programowania na URL-ach.</b></li> <li>3. <b>Zaawansowane technologie sieci lokalnych. Segmentacja sieci lokalnych. Rozpinanie LAN-u na wielu switchach. Protokół STP (Spanning Tree Protocol). Wirtualne LAN-y, czyli technologia VLAN: statyczne sieci wirtualne, dynamiczne sieci wirtualne.</b></li> <li>4. <b>Bezpieczeństwo w warstwie transportu. Bezpieczeństwo na WWW. Secure Socket Layer (SSL) oraz Transport Layer Security (TLS). Secure Electronic Transaction (SET).</b></li> <li>5. <b>Bezprzewodowe technologie w sieciach lokalnych. WLAN IEEE 802.11. Bezpieczeństwo i protokół WEP.</b></li> <li>6. <b>Zarządzanie sieciami, część 1. Wprowadzenie do zarządzania sieciami, motywacje, główne komponenty. Infrastruktura do zarządzania sieciami (internetowymi). MIB: management information base, SMI: Structure of Management Information - język do definiowania struktur danych.</b></li> <li>7. <b>Zarządzanie sieciami, część 2. SNMP: protokół do zarządzania sieciami. Bezpieczeństwo i administrowanie sieciami. Zdalny nadzór sieci – RMON. Uniwersalne sposoby prezentacji danych: ASN.1 - Abstract Syntax Notation.</b></li> <li>8. <b>Bezpieczeństwo w warstwie sieci. IP Security. Wirtualne sieci prywatne VPN.IP Mobile.</b></li> <li>9. <b>Zaawansowane technologie w warstwie Sieci. Multicasting, adresacja, routing.</b></li> <li>10. <b>ATM - kompletna technologia sieciowa w roli sieci szkieletowej Internetu. Warstwy w ATM. Struktura komórki ATM. IP nad ATM.</b></li> <li>11. <b>Sieci rozległe. Technologie stosowane w sieciach WAN, urządzenia wykorzystywane w WAN. Korelacja pomiędzy WAN a modelem OSI/ISO, protokoły enkapsulacji w WAN.</b></li> <li>12. <b>ISDN. Wprowadzenie do ISDN. Korelacja z modelem OSI/ISO, Tryby BRI i PRI. Konfiguracja ISDN, Routing Dial-on-Demand.</b></li> <li>13. <b>Frame Relay. Wprowadzenie do Frame Relay, definicje podstawowych pojęć związanych z FR. Omówienie LMI (implementacji FR firmy Cisco).</b></li> <li>14. <b>Pozostałe protokoły w sieciach WAN. PPP, xDSL, SDLC (HDLC), X.25.</b></li> </ol>		

15. **Omówienie najnowszych technologii sieciowych. Przegląd nowych rozwiązań i technologii w sieciach komputerowych. Zakreślenie trendów i perspektyw rozwoju w sieciach komputerowych.**

Treści zajęć laboratoryjnych

1. **Realizacja aplikacji w oparciu o protokół UDP. Realizacja pewnego przesyłania w oparciu o protokół UDP.**
2. **Programowanie na protokole HTTP. Programowanie aplikacji w oparciu o adresy URL.**
3. **Segmentacja w sieciach LAN, tworzenie sieci VLAN. Sposoby przełączania, protokół Spanning Tree. Konfiguracja usług DHCP na routerach.**
4. **Programowanie na gniazdach SSL - cz. 1.** Generowanie kluczy, magazynów i certyfikatów. Realizacja prostej aplikacji klient serwer z wykorzystaniem gniazd SSL (przy ustawieniach domyślnych).
5. Konfiguracja bezprzewodowych urządzeń sieciowych. Realizacja bezpieczeństwa w sieciach bezprzewodowych. Konfigurowanie NAT.
6. **Programowanie na gniazdach SSL - cz. 2.** Konstruowanie gniazd SSL w sposób jawny. Realizacja aplikacji z uwierzytelnianiem klienta i serwera. Realizacja aplikacji z wykorzystaniem protokołu HTTPS.
7. **Programowanie z wykorzystaniem protokołu SNMP.** Realizacja aplikacji odpytującej urządzenia w sieci poprzez agentów SNMP. Realizacja aplikacji wprowadzającej zmiany dla wartości wybranych identyfikatorów obiektów bazy MIB. Prosta aplikacja wysyłająca i odbierająca pułapki.
8. **Protokoły routingu (cz. 1).** Konfiguracja protokołów wektora odległości RIP v 2, EIGRP.
9. **Protokoły routingu (cz. 2).** Konfiguracja protokołu stanu łącza OSPF, redystrybucja pomiędzy protokołami routingu.
10. **Listy dostępu.** Używanie, przetwarzanie i wykorzystanie list dostępu do filtrowania pakietów IP. **Konfigurowanie standardowych i rozszerzonych list dostępu w routerach.**
11. **Aplety - cz. 1. Architektura apletu. Budowa prostego apletu, proste metody wyświetlania składników apletu.**
12. **Aplety - cz. 2.** Obsługa zdarzeń, klasy obsługi zdarzeń, interfejsy nasłuchujące zdarzeń. Wprowadzenie do AWT (Abstract Window Toolkit).
13. **Podstawowa konfiguracja i weryfikacja Frame Relay.** Realizacja topologii Frame Relay z wykorzystaniem DLCI i LMI.
14. Konfiguracja uwierzytelniania w PPP (PAP, CHAP). Komunikacja w protokole PPP, konfiguracja DDR. Wykorzystanie ramek LCP i NCP w protokole PPP.
15. Testowanie poprawności działania sieci, diagnozowanie i usuwanie usterek w sieciach komputerowych. Prezentacja i obrona zadań indywidualnych.

**Literatura podstawowa:**

1. Kurose J. F., Ross K. W., Sieci komputerowe. Od ogółu do szczegółu z Internetem w tle, Wydawnictwo Helion, Gliwice 2006
2. K. Krysiak. Sieci Komputerowe - Kompendium. Wydawnictwo Helion 2005
3. T. Sheldon. Wielka Encyklopedia Sieci Komputerowych. Wydawnictwo Robomatic s.c. 1999.

**Literatura dodatkowa:**

1. Akademia Sieci Cisco. CCNA Exploration, Semestr 1. PWN, Warszawa 2011
2. Leinwand, B. Pinsky. Konfiguracja Routerów Cisco. Podstawy. Mikom, Warszawa 2002.
3. M. Sportack. Routing IP - podstawowy podręcznik. Mikom, Warszawa 2000.
4. M. Sportack. Sieci komputerowe. Wydawnictwo Helion 2004,
5. R. Wright. Elementarz routingu IP. Mikom, Warszawa 1999.

**Planowane formy/działania/metody dydaktyczne:**

Wykład tradycyjny wspomagany technikami multimedialnymi, Laboratoria z wykorzystaniem sprzętu sieciowego. Zamieszczanie na stronach internetowych zadań i materiałów do ćwiczeń.

**Sposoby weryfikacji efektów kształcenia osiągniętych przez studenta:**

Efekt U\_01 jest sprawdzany przy obronie programistycznego zadania indywidualnego.

Efekty U\_02 – U\_04, są sprawdzane w czasie ocenianych zadań na laboratoriach, niektóre zadania wykonywane są w grupach i w ten sposób sprawdzane jest efekt K\_02.

Efekty W\_01 – W\_05 i K\_01, sprawdzane są egzaminie.

**Forma i warunki zaliczenia:**

Moduł kończy się egzaminem. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie laboratorium.

Na zaliczenie laboratorium składają się oceny cząstkowe uzyskane na regularnych zajęciach z nauczycielem akademickim oraz z samodzielnie wykonanego zadania indywidualnego według schematu:

- Regularne zajęcia – 26 pkt.,
- Obrona zadania indywidualnego – 14 pkt.

Zajęcia laboratoryjne będą zaliczone w wypadku uzyskania co najmniej połowy punktów z poszczególnych form aktywności studenta: regularne zajęcia – co najmniej 13 pkt., obrona indywidualnego zadania – co najmniej 7 pkt.

Na tej formie zajęć student może maksymalnie uzyskać 40 pkt.

Egzamin jest egzaminem pisemnym. Można na nim uzyskać do 60 pkt. Egzamin będzie zaliczony w przypadku uzyskania co najmniej 30 pkt. Ocena końcowa z modułu (wystawiana po zaliczeniu wszystkich części składowych), w zależności od sumy uzyskanych punktów (maksymalnie 100pkt.) jest następująca (w nawiasach ocena wg skali ECTS):

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

**Bilans punktów ECTS:****Studia stacjonarne**

Aktywność	Obciążenie studenta
Udział w wykładach	30 godz.
Udział w ćwiczeniach laboratoryjnych	45 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	25 godz.
Udział w konsultacjach godz. z przedmiotu	5 godz.

Przygotowanie się do egzaminu i obecność na egzaminie	20 godz.
Sumaryczne obciążenie pracą studenta	125 godz.
Punkty ECTS za przedmiot	5 ECTS
Studia niestacjonarne	
Aktywność	Obciążenie studenta
Udział w wykładach	18 godz.
Udział w ćwiczeniach laboratoryjnych	24 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	51 godz.
Udział w konsultacjach godz. z przedmiotu	2 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	30 godz.
Sumaryczne obciążenie pracą studenta	125 godz.
Punkty ECTS za przedmiot	5 ECTS



<b>Sylabus przedmiotu / modułu kształcenia</b>		
<b>Nazwa przedmiotu/modułu kształcenia:</b>	<b>Rozproszone systemy operacyjne</b>	
<b>Nazwa w języku angielskim:</b>	Distributed operating systems	
<b>Język wykładowy:</b>	polski	
<b>Kierunek studiów, dla którego przedmiot jest oferowany:</b>	informatyka	
<b>Jednostka realizująca:</b>	Wydział Nauk Ścisłych i Przyrodniczych	
<b>Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):</b>	fakultatywny	
<b>Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):</b>	pierwszego stopnia	
<b>Rok studiów:</b>	trzeci	
<b>Semestr:</b>	szósty	
<b>Liczba punktów ECTS:</b>	4	
<b>Imię i nazwisko koordynatora przedmiotu:</b>	dr Piotr Świtalski	
<b>Imię i nazwisko prowadzących zajęcia:</b>	dr Piotr Świtalski	
<b>Założenia i cele przedmiotu:</b>	<p>Celem przedmiotu jest zaznajomienie studentów z zagadnieniami występującymi w rozproszonych systemach operacyjnych w stopniu wystarczającym do ich zrozumienia.</p> <p>W założeniach do tego przedmiotu przewiduje się zajęcia praktyczne z użyciem komputerów, podczas których studenci nabędą umiejętności programowania w rozproszonych systemach operacyjnych. Dodatkowo przedmiot zakłada, że studenci nabędą kompetencje w zakresie pracy w zespole.</p>	
<b>Symbol efektu</b>	<b>Efekt uczenia się: WIEDZA</b>	<b>Symbol efektu kierunkowego</b>
<b>W_01</b>	Zna i rozumie w pogłębionym stopniu zagadnienia związane z rozproszonymi systemami operacyjnymi	<b>K_W05, K_W07, K_W15</b>
<b>W_02</b>	Zna i rozumie w pogłębionym stopniu zagadnienia związane z programowaniem mechanizmów stosowanych w rozproszonych systemach operacyjnych	<b>K_W06, K_W07, K_W15</b>

<b>W_03</b>	Zna i rozumie w pogłębionym stopniu mechanizmy wykorzystywane do koordynacji rozproszonych systemów operacyjnych	<b>K_W07, K_W15</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: UMIEJĘTNOŚCI</b>	<b>Symbol efektu kierunkowego</b>
<b>U_01</b>	Potrafi zaimplementować mechanizmy związane z koordynacją procesów w rozproszonych systemach operacyjnych	<b>K_U11, K_U13, K_U18, K_U26</b>
<b>U_02</b>	Potrafi zaprogramować symulator systemu rozproszonej komunikacji procesów	<b>K_U12, K_U13, K_U26</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: KOMPETENCJE SPOŁECZNE</b>	<b>Symbol efektu kierunkowego</b>
<b>K_01</b>	Jest gotów do uczestnictwa w zespole programistycznym, w którym zespół wykorzystuje mechanizmy rozproszonych systemów operacyjnych. Student w świadomy sposób będzie w stanie określić swoją rolę w takim zespole i wskazać na konsekwencje swoich działań.	<b>K_K01, K_K04</b>
<b>Forma i typy zajęć:</b>	Studia stacjonarne: wykłady (30 godz.), ćwiczenia laboratoryjne (45 godz.) Studia niestacjonarne: wykłady (18 godz.), ćwiczenia laboratoryjne (24 godz.)	
<b>Wymagania wstępne i dodatkowe:</b>		
Warunkiem uczestnictwa w zajęciach jest znajomość zagadnień z przedmiotu systemy operacyjne lub znajomość literatury obowiązującej w tym przedmiocie.		
<b>Treści modułu kształcenia:</b>		
<ol style="list-style-type: none"> <li><b>Struktury rozproszonych systemów operacyjnych.</b> Podstawy: sieciowe systemy operacyjne i rozproszone systemy operacyjne. Motywy: dzielenie zasobów, przyspieszanie obliczeń, niezawodność. Obecne architektury.</li> <li><b>Topologie sieci komputerowych.</b> Topologia: sieci całkowicie i częściowo połączone, sieci hierarchiczne i gwiazdowe. Topologia sieci pierścieniowych i z szynami wielodostępnymi, sieci mieszane. Typy sieci: sieci lokalne i rozległe.</li> <li><b>Komunikacja: nazewnictwo i tłumaczenie nazw.</b> Strategie wyboru trasy, postępowanie z pakietami. Strategie projektowe: model ISO, model TCP/IP. Przykłady działania sieci.</li> <li><b>Sieciowe i rozproszone systemy operacyjne.</b> Sieciowe systemy operacyjne: zdalna rejestracja, przesyłanie odległych plików. Rozproszone systemy operacyjne: wędrówka danych, wędrówka obliczeń, wędrówka procesów.</li> <li><b>Usługi zdalne.</b> Zdalne wywołania procedur, wątki. Odporność: wykrywanie uszkodzeń, rekonfiguracja.</li> <li><b>Interfejs systemu plików.</b> Pliki: atrybuty, operacje, typy, struktury. Metody dostępu: sekwencyjny, bezpośredni, inne. Struktura katalogu: katalog jednopoziomowy, dwupoziomowy, o strukturze drzewiastej.</li> <li><b>Rozproszone systemy plików.</b> Nazewnictwo i przejrzystość: struktury i schematy nazewnictwa, techniki implementacji. Zdalny dostęp do plików: przechowywanie podręczne,</li> </ol>		

uaktualnianie, spójność, obsługa zdalna. Obsługa doglądana i niedoglądana. Zwiokrotnianie pliku.

8. **Zakleszczenia.** Charakterystyka zakleszczenia. Metody postępowania z zakleszczeniami. Zapobieganie zakleszczeniom. Unikanie zakleszczeń.
9. **Koordinacja rozproszona.** Porządkowanie zdarzeń. Wzajemne wykluczanie. Niepodzielność. Sterowanie współbieżnością. Protokoły blokowania zasobów. Zastosowanie znaczników czasu.
10. **Równoważenie obciążenia.** Procesy i wątki. Migracja procesów. Algorytmy równoważenia obciążenia: Round robin DNS. Delegowanie DNS. Request Counting. Agregacja łączy.
11. **Wirtualizacja i przetwarzanie w chmurze.** Hipernadzorcy. Wirtualizacja pamięci i wejścia-wyjścia. Chmury obliczeniowe.
12. **Wielokomputery.** Niskopoziomowe oprogramowanie komunikacyjne. Rozproszona pamięć współdzielona. Równoważenie obciążenia.
13. **Systemy gridowe.** Struktura i sposób działania. Implementacje – Unicore, Condor, Globus.
14. **Systemy klastrowe.** Typy klastrów. Implementacje – MOSIX, SSI, Kerrighed, kolejkowy PBS.
15. **Bezpieczeństwo systemów rozproszonych.** Pojęcia. Izolacja usług. Kontrola dostępu. Zarządzanie zaufaniem. Protokoły bezpieczeństwa.

#### Literatura podstawowa:

1. Stallings W.: Systemy operacyjne. Architektura, funkcjonowanie i projektowanie. Wydanie IX, Helion 2018.
2. Silberschatz A., Galvin P. B.: Podstawy systemów operacyjnych, wydanie VI, WNT, 2005.

#### Literatura dodatkowa:

1. Tanenbaum A. S., Bos H.: Systemy operacyjne. Wydanie IV, Helion, 2015.
2. Switalski P., Seredynski F.: Multiprocessor Scheduling by Generalized Extremal Optimization, Journal of Scheduling: Volume 13, Issue 5 (2010), Springer, 2010.

#### Planowane formy/działania/metody dydaktyczne:

Wykład tradycyjny wspomagany technikami multimedialnymi, zajęcia laboratoryjne z użyciem komputera. Problemy i zadania ćwiczeniowe zamieszczane są na stronie internetowej prowadzącego zajęcia.

#### Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:

Efekty W\_01 do W\_03 weryfikowane będą poprzez egzamin pisemny, a także w toku weryfikacji przygotowania do kolejnych zajęć laboratoryjnych. Na egzaminie pytania będą dotyczyły poznanej wiedzy dotyczącej rozproszonych systemów operacyjnych. Przykładowe pytania:

- Na czym polega zdalne wywołanie procedury?
- Czym jest zakleszczenie? Jak zapobiec zakleszczeniom?
- Przedstaw podstawowe cechy koordynacji rozproszonej.

Przed egzaminem studenci będą mieli dostęp do przykładowych pytań na egzamin.

Efekty U\_01 - U\_02 będą sprawdzane systematycznie na zajęciach laboratoryjnych. Przykładowe zadania:

- Skonfiguruj i uruchom serwer OpenLDAP zaczynając od pustej bazy informacji katalogowej.

- Skonfiguruj i uruchom serwer Samba udostępniający przykładowy katalog do zapisu. Zweryfikuj możliwość dokonywania modyfikacji w tym katalogu.
- Stwórz aplikację udostępniającą dwuargumentową funkcję rkill() umożliwiającą wysłanie dowolnego sygnału do dowolnego procesu.

Materiały na następne laboratorium będą dostępne na dwa dni przed zajęciami.

Studenci będą również musieli wykonać zadanie indywidualne. Przykładowym zadaniem jest:

- Stworzyć działający symulator sieci o topologii pierścienia przekazujący znacznik i umożliwiający nadanie i odebranie komunikatu. Należy zaimplementować najważniejsze elementy ramki FDDI oraz jeden z algorytmów:
  - algorytm czasowego posiadania znacznika,
  - algorytm procesu uzgadniania znacznika.

Efekt K\_01 będzie weryfikowany w oparciu o posiadaną wiedzę i umiejętności w czasie zajęć laboratoryjnych oraz podczas realizacji zadania projektowego.

### Forma i warunki zaliczenia:

Ocena z przedmiotu składa się z trzech ocen cząstkowych:

- oceny z zajęć laboratoryjnych,
- oceny z projektu zespołowego,
- oceny z egzaminu końcowego.

Na ocenę z zajęć laboratoryjnych składają się oceny cząstkowe uzyskane na regularnych zajęciach z nauczycielem akademickim, za które można uzyskać sumarycznie 24 pkt. Dodatkowo student jest zobowiązany do realizacji projektu zespołowego, z którego może otrzymać maksymalnie 16 pkt. Zaliczenie zajęć laboratoryjnych możliwe jest po pozytywnie zakończonej obronie projektu zespołowego i uzyskaniu sumarycznie co najmniej 51% liczby punktów z tej formy zaliczenia.

W trakcie sesji odbędzie się egzamin końcowy. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie laboratorium. Egzamin przewidziany jest w formie pisemnej. Można na nim uzyskać maksymalnie 60 pkt. Egzamin będzie zaliczony w przypadku uzyskania co najmniej 51% liczby punktów z tej formy zaliczenia. Ocena końcowa z przedmiotu, w zależności od sumy uzyskanych punktów (maksymalnie 100 pkt) jest następująca (w nawiasach ocena wg skali ECTS):

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

### Bilans punktów ECTS:

Studia stacjonarne

Aktywność

Obciążenie studenta

Udział w wykładach	30 godz.
Udział w ćwiczeniach laboratoryjnych	45 godz.
Przygotowanie projektu indywidualnego	10 godz.
Przygotowanie się do egzaminu	8 godz.
Przygotowanie się do ćwiczeń laboratoryjnych	5 godz.
Udział w konsultacjach z przedmiotu	2 godz.
<b>Sumaryczne obciążenie pracą studenta</b>	<b>100 godz.</b>
<b>Punkty ECTS za przedmiot</b>	<b>4 ECTS</b>
Studia niestacjonarne	
Aktywność	Obciążenie studenta
Udział w wykładach	18 godz.
Udział w ćwiczeniach laboratoryjnych	24 godz.
Przygotowanie projektu indywidualnego	10 godz.
Przygotowanie się do egzaminu	22 godz.
Przygotowanie się do ćwiczeń laboratoryjnych	24 godz.
Udział w konsultacjach z przedmiotu	2 godz.
<b>Sumaryczne obciążenie pracą studenta</b>	<b>100 godz.</b>
<b>Punkty ECTS za przedmiot</b>	<b>4 ECTS</b>

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:		Projektowanie i administrowanie sieciami komputerowymi
Nazwa w języku angielskim:		Design and Administration of Computer Networks
Język wykładowy:	polski	
Kierunek studiów, dla którego przedmiot jest oferowany:		informatyka
Jednostka realizująca:	Wydział Nauk Ścisłych i Przyrodniczych	
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):		fakultatywny
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):		pierwszego stopnia
Rok studiów:	trzeci	
Semestr:	szósty	
Liczba punktów ECTS:	4	
Imię i nazwisko koordynatora przedmiotu:		Prof. dr hab. Stanisław Ambroszkiewicz
Imię i nazwisko prowadzących zajęcia:		mgr Zbigniew Młynarski
Założenia i cele przedmiotu:		Celem modułu jest zapoznanie z zasadami projektowania i administrowania sieciami komputerowymi
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	Student zna i rozumie podstawy projektowania i administrowania sieciami komputerowymi, a w szczególności projektowania i administrowania sieciami TCP/IP.	K_W07
W_02	Zna i rozumie główne zasady fizycznego i logicznego projektu sieci kampusowych i administrowania hostami.	K_W07
W_03	Zna i rozumie sposoby monitorowania sieci komputerowych i administrowania sieciowymi systemami plików NFS i CIFS,	K_W07
W_04	Zna i rozumie usługi nazewnicze i katalogowe w systemach heterogenicznych oraz sieci pamięci masowych SAN.	K_W07
W_05	Zna i rozumie zasadę działania platformy wirtualizacji hostów i serwerów.	K_W07
Symbol efektu	Efekt uczenia się: UMIEJĘTNOŚCI	Symbol efektu kierunkowego
U_01	Potrafi zaprojektować i administrować siecią komputerową.	K_U10
U_02	Potrafi sporządzać projekty logiczne i fizyczne sieci komputerowych z wykorzystaniem narzędzi projektowych,	K_U10, K_U11
U_03	Potrafi administrować sieciami TCP/IP i konfigurować routing.	K_U10

<b>U_04</b>	Potrafi korzystać z narzędzi SNMP w zarządzaniu sieciami i konfigurować serwery usług sieciowych.	<b>K_U10</b>
<b>U_05</b>	Potrafi formułować opinie na temat podstawowych zagadnień z projektowania i administrowania sieciami komputerowymi	<b>K_U10</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: KOMPETENCJE SPOŁECZNE</b>	<b>Symbol efektu kierunkowego</b>
<b>K_01</b>	Jest gotów do podejmowania decyzji i krytycznej oceny własnych rozwiązań w rozwiązywaniu problemów w administrowaniu sieciami komputerowymi	<b>K_K01</b>
<b>Forma i typy zajęć:</b>	studia stacjonarne: wykłady (30 godz.), ćwiczenia laboratoryjne (45 godz.) studia niestacjonarne: wykłady (18 godz.), ćwiczenia laboratoryjne (24 godz.)	
<b>Wymagania wstępne i dodatkowe:</b>		
Znajomość podstawowych pojęć i umiejętności z zakresu modułów: <ol style="list-style-type: none"> <li>1. Architektura komputerów,</li> <li>2. Systemy operacyjne</li> <li>3. Technologie sieciowe</li> </ol>		
<b>Treści modułu kształcenia:</b>		
Treści wykładu: <ol style="list-style-type: none"> <li>1. <b>Fizyczny projekt sieci.</b> Wybór technologii i urządzeń dla sieci kampusowych, wybór technologii i urządzeń dla sieci przedsiębiorstwa</li> <li>2. <b>Logiczny projekt sieci.</b> Projektowanie topologii sieci, opracowywanie modeli adresowania i nazewnictwa, wybieranie protokołów przełączania i routingu, tworzenie strategii bezpieczeństwa sieciowego.</li> <li>3. <b>Testowanie, optymalizacja i dokumentowanie projektu sieci.</b> Testowanie projektu sieci, optymalizacja projektu sieci, dokumentowanie projektu sieci.</li> <li>4. <b>System operacyjny jako komponent administrowanej sieci.</b> Sieciowe systemy operacyjne, systemy plików, procesy i sterowanie zadaniami.</li> <li>5. <b>Administrowanie hostami.</b> Proces instalowania oprogramowania, projektowanie infrastruktury serwerowni i centrum danych, uruchamianie i zamykanie systemów, konfiguracja i personalizacja stacji roboczych.</li> <li>6. <b>Zarządzanie i monitorowanie sieci.</b> Zarządzanie desktopami, zarządzanie adaptacyjne, zdalne sterowanie.</li> <li>7. <b>Zarządzanie tożsamością.</b> Polityka obsługi kont, środowisko rejestracji się w systemie.</li> <li>8. <b>Administrowanie rozproszonymi systemami plików.</b> Sieciowy system plików NFS, system plików CIFS.</li> <li>9. <b>Usługi nazewnicze i katalogowe.</b> Modele i zasady administrowania sieciami, usługa nazewnicza DNS, model informacyjny a usługi katalogowe.</li> <li>10. <b>Zarządzanie konfiguracją oraz utrzymanie sieci.</b> Konfiguracja sieci i reguły polityk, zarządzanie zmianami konfiguracyjnymi, narzędzia do automatyzacji administrowania.</li> <li>11. <b>Zarządzanie awariami.</b> Tolerowanie awarii i propagacja uszkodzeń, analiza schematu zdarzeń, monitorowanie sieci komputerowych.</li> <li>12. <b>Zabezpieczanie i archiwizacja danych.</b> Backup i archiwizacja, rodzaje pamięci masowych,</li> </ol>		

13. **Administrowanie sieciami bezprzewodowymi.** Standaryzacja bezprzewodowych sieci WLAN, bezprzewodowe sieci metropolitalne
14. **Sieci pamięci masowych.** Architektury pamięci masowych, przełączniki SAN, zarządzanie w sieciach SAN
15. **Emulacja i wirtualizacja.** Platformy wirtualizacji hostów, wykorzystanie wirtualizacji w administrowaniu sieciami, technologie wirtualizacji serwerów, mechanizmy ochrony wirtualizowanych zasobów.

Treści zajęć laboratoryjnych:

1. **Narzędzia do planowania i projektowania sieci komputerowych.** Oprogramowanie Microsoft Visio, biblioteki urządzeń sieciowych (Netformx DesignXpert), alternatywne rozwiązania Open Source (Dia).
2. **Projekt fizyczny sieci kampusowej.** Przykład projektu sieci kampusowej, charakterystyka ruchu aplikacji sieciowych, przeprojektowanie sieci kampusowej.
3. **Logiczny projekt sieci.** Przygotowanie topologii projektu sieci kampusowej, projektowanie modelu nazewnictwa.
4. **Dokumentowanie projektu sieci.** Odpowiadanie na zapytanie ofertowe klienta, przygotowywanie zawartości dokumentacji projektu.
5. **Wprowadzenie do administrowania sieciami TCP/IP.** Podstawowe polecenia administracyjne, monitorowanie protokołów za pomocą analizatora Wireshark, polecenia ifconfig oraz ip.
6. **Konfiguracja routowania.** Podstawowe konfiguracje routowania, budowanie statycznej tablicy routowania.
7. **Technologie zarządzania sieciami komputerowymi.** Czynności pomiarowe w sieciach komputerowych, pakiety wykorzystujące narzędzia SNMP.
8. **Usługa przesyłania plików.** Klient FTP, instalacja i podstawowa konfiguracja serwera FTP, zaawansowana konfiguracja serwera proFTPd.
9. **Konfiguracja usługi DHCP.** Konfiguracja klienta i serwera DHCP.
10. **Dzielenie plików.** Instalacja NFS, konfiguracja serwera i klienta NFS,
11. **Administrowanie usługami nazwicznymi.** Konfiguracja hosta, opcje konfiguracyjne programu nslookup.
12. **Konfiguracja serwera nazw DNS.** Konfiguracja BINDa, konfiguracja demona named.
13. **Konfiguracja serwera pocztowego.** Konfiguracja programu sendmail, konfiguracja POP i IMAP, uruchamianie demonów POP i IMAP, wykorzystywanie fetchmaila.
14. **Administrowanie heterogenicznymi sieciami komputerowymi.** Narzędzia do zarządzania w środowiskach systemów UNIX, Linux i Windows, jednorodny dostęp do zasobów,
15. **Prezentacja indywidualnego projektu.** Zaliczenie przedmiotu. prezentacja wyników prac, dyskusja i ocena projektów indywidualnych.

#### Literatura podstawowa:

1. Bauer K., Automating UNIX and Linux Administration, Apress, 2003
2. Bruno A., Kim J., CCDA. Certyfikat projektanta sieci Cisco, Wydawnictwo MIKOM, Warszawa 2004
3. Burgess M., Principles of Network and System Administration, Wiley, 2004
4. Hunt C.: TCP/IP - Administracja sieci, Wydawnictwo RM, Warszawa 2003
5. Oppenheimer P., Projektowanie sieci metoda Top-Down, Wydawnictwo Naukowe PWN, Warszawa 2006

#### Literatura dodatkowa:

1. Davies J., Whittaker R., von Hagen W., SUSE Linux 10 Bible, Wiley 2006
2. Krysiak K.: Sieci komputerowe. Kompendium, Wydawnictwo Helion, Gliwice 2005



3. Pawlak R., Okablowanie strukturalne sieci, Wydawnictwo Helion, Gliwice 2006
4. Tanenbaum A. S., Van Steen M., Systemy rozproszone. Zasady i paradygmaty, WNT, Warszawa 2006

#### Planowane formy/działania/metody dydaktyczne:

Wykład tradycyjny wspomagany technikami multimedialnymi, Laboratoria z wykorzystaniem sprzętu sieciowego. Zamieszczanie na stronach internetowych zadań i materiałów do ćwiczeń.

#### Sposoby weryfikacji efektów kształcenia osiągniętych przez studenta:

Efekty U\_01-U\_05 sprawdzane są na ocenianych zajęciach podczas laboratorium i podczas obrony zadania indywidualnego. Efekty W\_01 – W\_05 sprawdzane będą na egzaminie pisemnym, efekty U\_01, K\_01 na egzaminie ustnym w sesji egzaminacyjnej.

#### Forma i warunki zaliczenia:

Moduł kończy się egzaminem. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie laboratorium.

Na zaliczenie laboratorium składają się oceny częściowe uzyskane na regularnych zajęciach z nauczycielem akademickim oraz z samodzielnie wykonanego zadania indywidualnego według schematu:

- Regularne zajęcia – 26 pkt.,
- Obrona zadania indywidualnego – 14 pkt.

Zajęcia laboratoryjne będą zaliczone w wypadku uzyskania co najmniej połowy punktów z poszczególnych form aktywności studenta: regularne zajęcia – co najmniej 13 pkt., obrona indywidualnego zadania – co najmniej 7 pkt.

Na tej formie zajęć student może maksymalnie uzyskać 40 pkt.

Egzamin jest egzaminem pisemnym. Można na nim uzyskać do 60 pkt. Egzamin będzie zaliczony w przypadku uzyskania co najmniej 30 pkt. Ocena końcowa z modułu (wystawiana po zaliczeniu wszystkich części składowych), w zależności od sumy uzyskanych punktów (maksymalnie 100pkt.) jest następująca (w nawiasach ocena wg skali ECTS):

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

#### Bilans punktów ECTS\*:

##### Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	30 godz.
Udział w ćwiczeniach laboratoryjnych	45 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	10 godz.

Udział w konsultacjach godz. z przedmiotu	5 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	10 godz.
Sumaryczne obciążenie pracą studenta	100 godz.
Punkty ECTS za przedmiot	4 ECTS
Studia niestacjonarne	
Aktywność	Obciążenie studenta
Udział w wykładach	18 godz.
Udział w ćwiczeniach laboratoryjnych	24 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	35 godz.
Udział w konsultacjach godz. z przedmiotu	3 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	20 godz.
Sumaryczne obciążenie pracą studenta	100 godz.
Punkty ECTS za przedmiot	4 ECTS

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:		Systemy i narzędzia bezpieczeństwa systemów i sieci komputerowych
Nazwa w języku angielskim:		Systems and tools for computer systems and networks security
Język wykładowy:	polski	
Kierunek studiów, dla którego przedmiot jest oferowany:		informatyka
Jednostka realizująca:	Wydział Nauk Ścisłych i Przyrodniczych	
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):		fakultatywny
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):		pierwszego stopnia
Rok studiów:	trzeci	
Semestr:	szósty	
Liczba punktów ECTS:	3	
Imię i nazwisko koordynatora przedmiotu:		dr Piotr Świtalski
Imię i nazwisko prowadzących zajęcia:		dr Piotr Świtalski
Założenia i cele przedmiotu:		Celem przedmiotu jest zaznajomienie studentów z problematyką bezpieczeństwa systemów komputerowych. Przedmiot ma za zadanie zapoznanie z klasycznymi technikami bezpieczeństwa w systemach komputerowych. W założeniach do tego przedmiotu przewiduje się zajęcia praktyczne z użyciem komputerów, podczas których studenci nabędą umiejętności z atakami na systemy komputerowe i również metody przeciwdziałania im. Ma również usystematyzować wiedzę w zakresie standardów bezpieczeństwa.
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	Zna i rozumie zagadnienia z zakresu bezpieczeństwa systemów komputerowych	K_W05, K_W07
W_02	Zna i rozumie standardy i uwarunkowania prawne związane z bezpieczeństwem systemów komputerowych	K_W04, K_W10

<b>W_03</b>	Zna i rozumie techniki związane z atakami na systemy komputerowe w tym związane z sieciami komputerowymi i potrafi im przeciwdziałać	<b>K_W05, K_W07</b>
<b>W_04</b>	Zna i rozumie mechanizmy bezpieczeństwa komputerowego i potrafi je wykorzystać w przedstawionym środowisku	<b>K_W07, K_W12, K_W15</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: UMIEJĘTNOŚCI</b>	<b>Symbol efektu kierunkowego</b>
<b>U_01</b>	Potrafi sprawnie wyszukiwać w literaturze informacje związane z bezpieczeństwem systemów komputerowych, potrafi wyszukać informacje na temat nowych podatności wykrytych w systemach komputerowych	<b>K_U01</b>
<b>U_02</b>	Potrafi sprawnie wykorzystać narzędzia bezpieczeństwa systemów komputerowych	<b>K_U02, K_U17</b>
<b>U_03</b>	Potrafi dobrać adekwatne zabezpieczenia w stosunku do zagrożenia w systemie komputerowym	<b>K_U10, K_U17, K_U25</b>
<b>Forma i typy zajęć:</b>	studia stacjonarne: wykłady (30 godz.), ćwiczenia laboratoryjne (30 godz.) studia niestacjonarne: wykłady (15 godz.), ćwiczenia laboratoryjne (15 godz.)	
<b>Wymagania wstępne i dodatkowe:</b>		
Warunkiem uczestnictwa w zajęciach jest uprzednie zaliczenie następujących przedmiotów: „Architektura systemów komputerowych”, „Systemy operacyjne”, „Technologie sieciowe” lub znajomość literatury obowiązującej w tych przedmiotach.		
<b>Treści modułu kształcenia:</b>		
<ol style="list-style-type: none"> <li><b>Koncepcja bezpieczeństwa komputerowego.</b> Podstawowe zasady. Właściwości i klasyfikacja koncepcji bezpieczeństwa: integralność, dostępność, poufność, niezaprzeczalność, odpowiedzialność. Podstawowe pojęcia. Model bezpieczeństwa sieciowego. Minimalne standardy bezpieczeństwa. Polityka bezpieczeństwa.</li> <li><b>Przestępczość komputerowa.</b> Przesłpstwa ujawnione i nieujawnione. Obiekty, typy i sprawcy przestępstw. Piractwo komputerowe, sabotaż, wywiad gospodarczy, szpiegostwo, przestępstwa bankowe. Inżynieria społeczna. Phising. Inżynieria odwrotna. Organizacje przeciwdziałające przestępczości komputerowej.</li> <li><b>Aspekty prawne ochrony informacji i normalizacja.</b> Stan prawny ochrony informacji w Polsce. Ochrona danych osobowych. Hacking. Prawo autorskie a programy komputerowe. Normalizacja w zakresie bezpieczeństwa systemów informatycznych.</li> <li><b>Charakterystyka zagrożeń w systemach informatycznych.</b> Specyfika systemów informatycznych. Klasyfikacja zagrożeń. Zagrożenia związane z DNS. Złośliwe oprogramowanie. Sieci botnet. Statystyka typowych zagrożeń.</li> <li><b>Klasyczne techniki szyfrowania.</b> Dziedzina kryptografii i podstawowe pojęcia. Podstawowe techniki szyfrowania: technika podstawieniowa, szyfr Cezara, szyfry mono i polialfabetyczne,</li> </ol>		

szyfr Playfaira, szyfr Vigenère'a. Techniki przestawieniowe, szyfr zygzakowy, maszyny wirnikowe. Szyfrowanie symetryczne. Ataki siłowe przeprowadzane na algorytmy szyfrowania.

6. **Szyfry strumieniowe i blokowe.** Szyfry strumieniowe. Szyfr strumieniowy RC4. Szyfry blokowe. Struktura i szyfr Feistela. Standard DES, efekt lawiny. Algorytm AES. Tryby operacyjne szyfrów blokowych. Szyfrowanie asymetryczne. Algorytm RSA.
7. **Podpis cyfrowy.** Idea podpisu cyfrowego. Wymagania stawiane podpisom cyfrowym. Mechanizm uwierzytelniania komunikatów. Kody uwierzytelnienia komunikatów MAC. Podpis cyfrowy ElGamal. Standard DSS. Algorytm DSA. Kryptograficzne funkcje haszujące. Algorytm SHA-512. Paradoks urodzin.
8. **Uwierzytelnianie.** Pojęcie uwierzytelniania. Uwierzytelnianie przez hasło. Strategie wyboru haseł. Inne metody uwierzytelniania. Protokoły uwierzytelniania: protokół challenge and response. Atak „człowiek pośrodku”. Dowód z wiedzą zerową. Uwierzytelnianie dwuskładnikowe. Hasło jednorazowe. Generowanie haseł jednorazowych – protokół S/KEY.
9. **Szkodliwe oprogramowanie.** Problemy związane z nieautoryzowanym dostępem do systemów komputerowych. Infekcje systemów komputerowych. Bezpieczny system operacyjny. Podatność systemów operacyjnych. Charakterystyka szkodliwego oprogramowania: tylne drzwi, bomby logiczne, konie trojańskie, wirusy, robaki, exploity i rootkity, keyloggery. Przeciwdziałanie szkodliwemu oprogramowaniu.
10. **Nadużycia w sieciach rozległych.** Protokoły komunikacyjne. Struktura nagłówek protokołów TCP/IP. Ataki na warstwę sieciową. Falszowanie pakietów IP. Fragmentacja pakietów IP. Ataki na warstwę transportową. Atak SYN Flood. Skanowanie portów. Ataki w sieciach komórkowych.
11. **Zapory sieciowe (firewalle).** Model ogólny zapory sieciowej. Charakterystyka firewalli. Ograniczenia firewalli. Firewall filtrujący pakiety. Firewall filtrujący pakiety z badaniem stanu pakietu. Brama aplikacyjna, brama transmisyjna. Implementacja firewalla. Strefa zdemilitaryzowana (DMZ). Przykładowa konfiguracja firewalla z DMZ.
12. **Systemy wykrywania intruzów.** Zachowania intruzywne. Wzorce zachowań intruzów. Wykrywanie intruzów. Statystyczna analiza zachowania. Wykrywanie intruzów w oparciu o reguły. Systemy IDPS. Audyt w systemach IDPS. Pułapki (Honeypoty).
13. **Bezpieczeństwo systemów internetowych.** Ataki w warstwie aplikacji. Przepelnienie bufora. Wstrzykiwanie kodu. Ataki SQL Injection. Błędy w uwierzytelnianiu użytkownika i zarządzania jego sesją. Atak XSS. Błędy w konfiguracji oprogramowania. Niewłaściwe zabezpieczenie wrażliwych danych. Atak Cross-Site Request Forgery (CSRF).
14. **Narzędzia bezpieczeństwa systemów komputerowych.** Testy penetracyjne. Program Nessus. Skaner Nikto. Narzędzia i skrypty badania serwisów WWW. Automatyczne narzędzia exploitów. Narzędzia do wstrzykiwania pakietów.
15. **Zapewnianie dostępności danych.** Utrzymanie ciągłości zasilania. Sposoby zapobiegania problemom zasilania, zasilacze awaryjne UPS. Ochrona danych przed utratą. Systemy macierzowe RAID. Kopie bezpieczeństwa.

#### Literatura podstawowa:

1. Stallings W.: Kryptografia i bezpieczeństwo sieci komputerowych. Matematyka szyfrów i techniki kryptologii, Wyd. Helion, Gliwice, 2012.
2. Prasad P.: Testy penetracyjne nowoczesnych serwisów. Kompendium inżynierów bezpieczeństwa, Wyd. Helion, Gliwice, 2017.
3. Stuttard D., Pinto M., The Web Application Hacker's Handbook, John Wiley & Sons, Inc, 2011.

4. Barczak A., Sydoruk T.: Bezpieczeństwo systemów informatycznych. Wyd. Akademii Podlaskiej, Siedlce, 2002.

#### Literatura dodatkowa:

1. Stallings W.: Kryptografia i bezpieczeństwo sieci komputerowych. Koncepcje i metody bezpiecznej komunikacji, Wyd. Helion, Gliwice, 2012.
2. Schetina E., Green K., Carlson J.: Bezpieczeństwo w sieci, Wyd. Helion, Gliwice, 2002.
3. Suehring S.: Zapory sieciowe w systemie Linux. Kompendium wiedzy o nftables. Wydanie IV, Wyd. Helion, Gliwice, 2015.

#### Planowane formy/działania/metody dydaktyczne:

Wykład tradycyjny wspomagany jest technikami multimedialnymi. Ćwiczenia laboratoryjne – zajęcia praktyczne z wykorzystaniem wybranych narzędzi programowych. Na stronie internetowej prowadzącego zamieszczone są materiały z problemami i zadaniami laboratoryjnymi.

#### Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:

Efekty W\_01 do W\_04 weryfikowane będą poprzez egzamin pisemny, a także w toku weryfikacji przygotowania do kolejnych zajęć laboratoryjnych. Na egzaminie pisemnym pytania będą dotyczyły poznanych technik ataków i sposobów zabezpieczania systemów komputerowych. Egzamin będzie również obejmował treści związane z kryptografią. Przykładowe pytania:

- Przedstaw schemat działania podpisu cyfrowego.
- Na czym polega uwierzytelnianie dwuskładnikowe?
- W jaki sposób przeprowadzany jest atak DDoS?
- Na czym polega szyfrowanie symetryczne?
- Czym są testy penetracyjne?

Przed egzaminem studenci będą mieli dostęp do przykładowych pytań na egzamin.

Efekty U\_01 do U\_03 będą sprawdzane systematycznie na zajęciach laboratoryjnych. Przykładowe zadania:

- Zaimplementuj certyfikat SSL i zaimplementuj go w wybranej przeglądarce internetowej.
- Napisz polecenie dla zapory iptables, które będzie odrzucało wszystkie nowe połączenia TCP z adresu IP o adresie 91.205.12.6 na porcie 22.

Materiały na następne laboratorium będą dostępne na dwa dni przed zajęciami.

#### Forma i warunki zaliczenia:

Ocena z przedmiotu składa się z dwóch ocen cząstkowych:

- oceny z zajęć laboratoryjnych,
- oceny z egzaminu końcowego.

Na ocenę z zajęć laboratoryjnych składają się oceny cząstkowe uzyskane na regularnych zajęciach z nauczycielem akademickim, za które można uzyskać sumarycznie 40 pkt. Zaliczenie zajęć laboratoryjnych możliwe po uzyskaniu co najmniej 51% liczby punktów z tej formy zaliczenia

Egzamin jest egzaminem pisemnym. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie laboratorium. Można na nim uzyskać maksymalnie 60 pkt. Egzamin będzie zaliczony w przypadku uzyskania co najmniej 51% liczby punktów z tej formy zaliczenia. Ocena końcowa z przedmiotu, w zależności od sumy uzyskanych punktów (maksymalnie 100 pkt) jest następująca (w nawiasach ocena wg skali ECTS):

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

#### **Bilans punktów ECTS:**

##### Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	30 godz.
Udział w ćwiczeniach laboratoryjnych	30 godz.
Przygotowanie się do egzaminu	8 godz.
Przygotowanie się do ćwiczeń laboratoryjnych	5 godz.
Udział w konsultacjach z przedmiotu	2 godz.
<b>Sumaryczne obciążenie pracą studenta</b>	<b>75 godz.</b>
<b>Punkty ECTS za przedmiot</b>	<b>3 ECTS</b>

##### Studia niestacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	15 godz.
Udział w ćwiczeniach laboratoryjnych	15 godz.
Przygotowanie się do egzaminu	20 godz.
Przygotowanie się do ćwiczeń laboratoryjnych	20 godz.
Udział w konsultacjach z przedmiotu	5 godz.
<b>Sumaryczne obciążenie pracą studenta</b>	<b>75 godz.</b>

<b>Punkty ECTS za przedmiot</b>	<b>3 ECTS</b>
---------------------------------	---------------



**Specjalność: Grafika komputerowa, semestr VI**

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:		Programowanie gier komputerowych
Nazwa w języku angielskim:		Computer game programming
Język wykładowy:	polski	
Kierunek studiów, dla którego przedmiot jest oferowany:		Informatyka
Jednostka realizująca:	Wydział Nauk Ścisłych i Przyrodniczych	
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):		fakultatywny
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):		pierwszego stopnia
Rok studiów:	trzeci	
Semestr:	szósty	
Liczba punktów ECTS:	5	
Imię i nazwisko koordynatora przedmiotu:		dr Mirosław Barański
Imię i nazwisko prowadzących zajęcia:		dr Mirosław Barański
Założenia i cele przedmiotu:		<p>Założono, że studenci poznają w odpowiednim stopniu zagadnienia związane z programowaniem gier komputerowych, w tym algorytmy w tym zakresie, środowiska programistyczne wspomagające programowanie gier komputerowych oraz dedykowane systemy informatyczne (głównie silniki).</p> <p>Studenci poznają zasady rządzące tworzeniem gier komputerowych: algorytmy, środowiska, popularne narzędzia oraz wybrane zagadnienia matematyczne i fizyczne wykorzystywane w tworzeniu gier.</p> <p>Celem kursu jest zapoznanie studentów zasadami tworzenia gier komputerowych, ze środowiskami do ich programowania oraz z wybranymi zagadnieniami fizyki związanymi z grami komputerowymi i zasadami tworzenia gier na urządzenia mobilne</p>
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	Zna i rozumie etapy tworzenia gier komputerowych	K_W06, K_W08
W_02	Zna i rozumie zasadę funkcjonowania środowisk i systemów informatycznych wspomagających tworzenie gier komputerowych	K_W06, K_W08
W_03	Zna i rozumie działania algorytmów wykorzystywanych w projektowaniu i implementacji gier komputerowych	K_W06, K_W08, K_W11
W_04	Zna i rozumie zaawansowane zagadnienia matematyki i fizyki związane z projektowaniem i implementacją gier komputerowych	K_W01, K_W02, K_W06

<b>W_05</b>	Zna i rozumie zasady tworzenia gier komputerowych na urządzenia mobilne	K_W06, K_W08
<b>Symbol efektu</b>	<b>Efekt uczenia się: UMIEJĘTNOŚCI</b>	<b>Symbol efektu kierunkowego</b>
<b>U_01</b>	Potrafi dobrać odpowiednie metody i narzędzia modelowania i programowania niezbędne do implementacji systemów gier komputerowych, potrafi je zaimplementować	K_U09, K_U10, K_U22
<b>U_02</b>	Potrafi dobrać odpowiednie środowiska, techniki i technologie do projektowania systemów gier komputerowych i wykorzystać je w implementacji	K_U11, K_U22
<b>U_03</b>	Potrafi porównać środowiska lub systemy grafiki komputerowej wybrane do projektowania systemów gier komputerowych ze względu na zadane kryteria użytkowe i ekonomiczne	K_U16
<b>U_04</b>	Potrafi ocenić przydatność rutynowych metod i narzędzi do rozwiązywania zadań związanych z projektowaniem gier komputerowych dotyczących akwizycji, projektowania i implementacji tych systemów	K_U17, K_U19
<b>Symbol efektu</b>	<b>Efekt uczenia się: KOMPETENCJE SPOŁECZNE</b>	<b>Symbol efektu kierunkowego</b>
<b>K_01</b>	jest gotów do krytycznej oceny posiadanej wiedzy oraz jest gotów do uznawania znaczenia wiedzy w rozwiązywaniu problemów poznawczych i praktycznych z zakresu informatyki	K_K01
<b>Forma i typy zajęć:</b>	Studia stacjonarne: wykłady (30 godz.), laboratoria (45 godz.), Studia niestacjonarne: wykłady (18 godz.), laboratoria (24 godz.),	
<b>Wymagania wstępne i dodatkowe:</b>		
1. Znajomość podstawowych pojęć i algorytmów grafiki komputerowej lub cyfrowego przetwarzania obrazu i dźwięku		
<b>Treści modułu kształcenia:</b>		
<ol style="list-style-type: none"> <li>1. Architektura silnika gry komputerowej</li> <li>2. Organizacja danych w grach komputerowych. Graf sceny. Podstawowe mechanizmy silników graficznych.</li> <li>3. Podsystem wyświetlania grafiki, eliminacja niewidocznych obiektów, renderowanie,</li> <li>4. Podsystem interakcji, wykrywanie i reakcja na kolizje,</li> <li>5. Programowanie ruchu wirtualnej postaci,</li> <li>6. Podsystem dźwięku,</li> <li>7. Zagadnienia sieciowe w grach komputerowych,</li> <li>8. Zaawansowane techniki graficzne, oświetlenie na wierzchołek i piksel, mapy nierówności.</li> <li>9. Dynamiczna kontrola poziomu szczegółowości sceny.</li> <li>10. Wykorzystanie Biblioteki XNA w grach komputerowych,</li> <li>11. Wykorzystanie silnika Unity w grach komputerowych,</li> <li>12. Wykorzystanie silnika Unreal Engine w grach komputerowych,</li> <li>13. Języki skryptowe w grach komputerowych,</li> <li>14. Programowanie gier komputerowych na urządzeniach mobilnych,</li> <li>15. Wydajność i optymalizacja gier komputerowych,</li> </ol>		
<b>Literatura podstawowa:</b>		
<ol style="list-style-type: none"> <li>1. Karl Bunyan, HTML5. Tworzenie gier z wykorzystaniem CSS i JavaScript (ebook), Helion 2016.</li> <li>2. Aram Cookson i inni, Unreal Engine w 24 godziny. Nauka tworzenia gier (ebook) Helion 2017.</li> <li>3. Jon Manning, Paris Buttfield-Addison, Unity. Tworzenie gier mobilnych (ebook), Helion 2018,</li> </ol>		

**Literatura dodatkowa:**

1. M. de Loura, Perełki programowania gier (tomy 1-3). Helion 2002.
2. Rafał Nowocień, Tworzenie gier komputerowych. Kompendium producenta, Helion 2019

**Planowane formy/działania/metody dydaktyczne:**

Wykład tradycyjny wspomagany technikami multimedialnymi. Zamieszczanie na stronach internetowych zadań i materiałów do laboratoriów.

**Sposoby weryfikacji efektów kształcenia osiągniętych przez studenta:**

Zakłada się znajomość przez studentów Unity i Unreal Engine na poziomie podstawowym – realizowane na kursie „Silniki gier komputerowych”. W tym celu zajęcia będą zsynchronizowane z tym kursem.

Efekt W\_01 będzie sprawdzany na egzaminie ustnym oraz podczas realizacji zadania indywidualnego. Na egzaminie będzie opisana modelowa gra komputerowa i jedną z zadań będzie wyszczególnienie oraz ogólna charakterystyka etapów projektowania gier komputerowych. Natomiast podczas realizacji zadania indywidualnego ten efekt będzie sprawdzany podczas analizy sprawozdania.

Efekt W\_02 będzie sprawdzany podczas egzaminu na egzaminie ustnym. Przykładowe pytanie to: Scharakteryzuj środowisko UNITY 3D.

Efekt W\_03 będzie sprawdzany na egzaminie ustnym. Przykładowe pytanie to: Omów problem wykrywania kolizji. Jak można go rozwiązać w implementacji gier komputerowych.

Efekt W\_04 będzie sprawdzany na egzaminie pisemnym i ustnym. Przykładowe pytanie to: Omów jakie zjawiska i jakie problemy występują podczas ruchu dwu kul.

Efekt W\_05 będzie sprawdzany na egzaminie ustnym. Przykładowe zadanie to: Omów proces tworzenia gry „Zgaduj-Zgadula”. Opisz metodę postępowania przy programowaniu tej gry na urządzenie mobilne w systemie Android.

Przed egzaminem studenci będą mieli dostęp do pełnej listy pytań na egzamin ustny oraz do przykładowych zadań na egzamin pisemny.

Efekty U\_01, U\_02, U\_03 i U\_04 będą sprawdzane na zajęciach laboratoryjnych i poprzez odpowiednio sformułowane zadanie indywidualne. Zajęcia laboratoryjne będą oceniane po każdym zajęciach. Zadania tam realizowane będą polegać na projektowaniu i implementacji gier komputerowych (prosty i złożonych w ramach zadania indywidualnego).

Zadania na następne laboratorium muszą być dostępne co najmniej tydzień przed zajęciami. Student, na podstawie podanej literatury, musi się do nich samodzielnie lub korzystając z konsultacji przygotować.

Efekt K\_01 będzie weryfikowany w czasie realizacji zadania indywidualnego. Zadania indywidualne będą realizowane w grupach 3-4 osobowych. Na wybranych zajęciach i konsultacjach studenci będą przedstawiali swoje rozwiązania. Na regularnych zajęciach będą one krytycznie oceniane przez całą grupę.

**Forma i warunki zaliczenia:**

Moduł kończy się egzaminem. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie laboratorium. Na zaliczenie laboratorium składają się oceny częściowe uzyskane na regularnych zajęciach z nauczycielem akademickim oraz z samodzielnie wykonanego zadania indywidualnego według schematu:

Regularne zajęcia – 26 pkt.,

Obrona zadania indywidualnego – 14 pkt.

Zajęcia laboratoryjne będą zaliczone w wypadku uzyskania co najmniej połowy punktów z poszczególnych form aktywności studenta: regularne zajęcia – co najmniej 13 pkt., obrona indywidualnego zadania – co najmniej 7 pkt. Na tej formie zajęć student może maksymalnie uzyskać 40 pkt.

Egzamin jest egzaminem ustnym. Można na nim uzyskać do 60 pkt. Egzamin będzie zaliczony w przypadku uzyskania co najmniej 30 pkt. Ocena końcowa z modułu, w zależności od sumy uzyskanych punktów (maksymalnie 100pkt.) jest następująca (w nawiasach ocena wg skali ECTS):

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

Poprawy:

Jednorazowa poprawa każdego kolokwium w trakcie zajęć w semestrze. Dwie poprawy obu kolokwium w sesji egzaminacyjnej, odpowiednio przed drugim i trzecim terminem egzaminu pisemnego.

Uwaga: Istnieje możliwość zwolnienia z egzaminu pisemnego lub ustnego studentów wyróżniających się na zajęciach laboratoryjnych. Warunkiem koniecznym zwolnienia z egzaminu jest uzyskanie 90% punktów możliwych do zdobycia w trakcie regularnych zajęć laboratoryjnych. Decyzję o ewentualnym zwolnieniu podejmuje osoba przeprowadzająca egzamin po zasięgnięciu opinii (poprzez rozmowę) osób prowadzących zajęcia. Decyzję o zwolnieniu prowadzący wykład przekazuje studentom nie później niż 2 tygodnie przed końcem semestru.

Bilans punktów ECTS:

Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	30 godz.
Udział w ćwiczeniach laboratoryjnych	45 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	25 godz.
Udział w konsultacjach godz. z przedmiotu	5 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	20 godz.

Sumaryczne obciążenie pracą studenta	125 godz.
Punkty ECTS za przedmiot	5 ECTS
Studia niestacjonarne	
Aktywność	Obciążenie studenta
Udział w wykładach	18 godz.
Udział w ćwiczeniach laboratoryjnych	24 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	51 godz.
Udział w konsultacjach godz. z przedmiotu	2 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	30 godz.
Sumaryczne obciążenie pracą studenta	125 godz.
Punkty ECTS za przedmiot	5 ECTS

<b>Sylabus przedmiotu / modułu kształcenia</b>		
<b>Nazwa przedmiotu/modułu kształcenia:</b>	<b>Akwizycja i przetwarzanie informacji wizualnej</b>	
<b>Nazwa w języku angielskim:</b>	Acquisition and processing of visual information	
<b>Język wykładowy:</b>	polski	
<b>Kierunek studiów, dla którego przedmiot jest oferowany:</b>	Informatyka	
<b>Jednostka realizująca:</b>	Wydział Nauk Ścisłych i Przyrodniczych	
<b>Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):</b>	fakultatywny	
<b>Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):</b>	pierwszego stopnia	
<b>Rok studiów:</b>	trzeci	
<b>Semestr:</b>	szósty	
<b>Liczba punktów ECTS:</b>	4	
<b>Imię i nazwisko koordynatora przedmiotu:</b>	dr Andrzej Salamończyk	
<b>Imię i nazwisko prowadzących zajęcia:</b>	dr Andrzej Salamończyk	
<b>Założenia i cele przedmiotu:</b>	Celem modułu jest zapoznanie z metodami pozyskiwania i przetwarzania informacji wizualnej.	
<b>Symbol efektu</b>	<b>Efekt uczenia się: WIEDZA</b>	<b>Symbol efektu kierunkowego</b>
<b>W_01</b>	Zna i rozumie funkcjonowanie urządzeń i metod akwizycji wizualnej.	<b>K_W08</b>
<b>W_02</b>	Zna i rozumie zasady działania urządzeń służących do wizualizacji.	<b>K_W08</b>
<b>W_03</b>	Zna i rozumie funkcjonowanie elementów składowych systemu automatycznego widzenia.	<b>K_W08</b>
<b>W_04</b>	Zna i rozumie zasady korzystania z przykładów deskryptorów kształtu.	<b>K_W08</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: UMIEJĘTNOŚCI</b>	<b>Symbol efektu kierunkowego</b>
<b>U_01</b>	Potrafi korzystać z narzędzi do tworzenia i przetwarzania grafiki rastrowej	<b>K_U11, K_U15, K_U17, K_U19</b>
<b>U_02</b>	Potrafi korzystać z narzędzi do tworzenia i przetwarzania grafiki wektorowej	<b>K_U11, K_U15, K_U17, K_U19</b>

<b>U_03</b>	Potrafi korzystać z narzędzi do przetwarzania sekwencji wideo.	<b>K_U11, K_U15, K_U17, K_U19</b>
<b>U_04</b>	Potrafi przetworzyć wstępnie obraz rastrowy w celu wykorzystania go w systemach rozpoznających.	<b>K_U09</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: KOMPETENCJE SPOŁECZNE</b>	<b>Symbol efektu kierunkowego</b>
<b>K_01</b>	Jest gotów do podejmowania decyzji dotyczącej wyboru odpowiednich rozwiązań i narzędzi oraz krytycznej oceny własnych rozwiązań	<b>K_K01</b>
<b>K_02</b>	Jest gotów do uznania znaczenia wiedzy dotyczącej zastosowań grafiki komputerowej oraz krytycznie potrafi ocenić swoje działania	<b>K_K01</b>
<b>Forma i typy zajęć:</b>	studia stacjonarne: wykłady (30 godz.), ćwiczenia laboratoryjne (45 godz.) studia niestacjonarne: wykłady (18 godz.), ćwiczenia laboratoryjne (24 godz.)	
<b>Wymagania wstępne i dodatkowe:</b>		
1. Znajomość podstawowych pojęć i algorytmów grafiki komputerowej (zakres przedmiotu Grafika komputerowa).		
<b>Treści modułu kształcenia:</b>		
<ol style="list-style-type: none"> <li>1. Percepcja informacji wizualnej. Źródła informacji wizualnej. Wzrok i wrażenia wizualne.</li> <li>2. Barwa w grafice komputerowej. Systemy barw. Kolorymetry.</li> <li>3. Akwizycja danych 2D. Fotografia i fotografie, mikroskopy, kamery. Obrazy HDR. Dyskretyzacja obrazu.</li> <li>4. Akwizycja danych 3D. Skanery. Tomografy.</li> <li>5. Potok graficzny 2D i 3D. Potok stały i programowalny. Teksturowanie.</li> <li>6. Urządzenia wizualizacyjne. Ekrany kineskopowe, ekrany plazmowe, ekrany elektroluminescencyjne. Projekторы. Holografia.</li> <li>7. Steroskopia. Tworzenie obrazów stereoskopowych. Technika anaglifowi i polaryzacyjna. Systemy aktywne i pasywne.</li> <li>8. Formaty plików w grafice komputerowej. Formaty rastrowe i wektorowe, stratne i bezstratne. Kompresja obrazów ruchomych.</li> <li>9. Wizja komputerowa. Elementy składowe systemu automatycznego widzenia.</li> <li>10. Przetwarzanie wstępne obrazu rastrowego. Odszumianie obrazu, filtracja, wyrównywanie histogramu, operacje morfologiczne.</li> <li>11. Narzędzia do przetwarzania grafiki rastrowej. Przegląd aplikacji. Praca w programach Photoshop i Gimp.</li> <li>12. Segmentacja obrazu. Detekcja krawędzi, szkieletyzacja. Przetwarzanie grafiki wektorowej.</li> <li>13. Narzędzia do przetwarzania grafiki wektorowej. Przegląd aplikacji. Praca w programach Corel i Adobe Illustrator.</li> <li>14. Ekstrakcja i selekcja cech. Deskryptory kształtu. Przegląd i ocena reprezentacji obiektów i ocena ich przydatności w systemach rozpoznających. Metody SIFT i PCA.</li> </ol>		



15. Automatyczna klasyfikacja obiektów. Metody klasyfikacji obiektów. Klasyfikatory odległościowe i statystyczne.

**Literatura podstawowa:**

6. W. S. Mokrzycki. Wprowadzenie do przetwarzania informacji wizualnej Tom 1 Percepcja akwizycja wizualizacja Exit 2010
7. W. S. Mokrzycki. Wprowadzenie do przetwarzania informacji wizualnej Tom 2 Dyskretyzacja obrazu, operacje pikselowe, morfologiczne Exit 2012
8. K. Stąpor. Metody klasyfikacji obiektów w wizji komputerowej. PWN 2011

**Literatura dodatkowa:**

6. R. S. Choraś. Komputerowa wizja. Metody interpretacji i identyfikacji obiektów. Exit 2005
7. M. Domański. Obraz cyfrowy Reprezentacja kompresja podstawy przetwarzania Standardy JPEG i MPEG. WKiŁ 2010.
8. J. D. Foley, A. van Dam, S. K. Feiner, J. F. Hughes. Wprowadzenie do grafiki komputerowej, WNT 1995 (lub wydanie późniejsze)Piotr Metzger. Anatomia PC. Wydanie XI. Helion 2007

**Planowane formy/działania/metody dydaktyczne:**

Wykład tradycyjny wspomagany technikami multimedialnymi. Zamieszczanie na stronach internetowych zadań i materiałów do laboratoriów.

**Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:**

Efekty U\_01 – U\_04 są sprawdzane w czasie ocenianych zadań na laboratoriach.

Efekty K\_01, K\_02 będą weryfikowane, w oparciu o posiadaną wiedzę i umiejętności w czasie zajęć laboratoryjnych, podczas egzaminu, a także będą sprawdzane podczas zadań problemowych podczas wykładów.

Efekty W\_01 – W\_04 sprawdzane są egzaminie.

Przykładowe pytania:

Efekt W\_01

Omów zasadę działania aparatu/kamery/skanera.

Efekt W\_02

Omów zasadę działania i główne cechy monitorów CRT/LCD/plazmowych.

Efekt W\_03

Wymień i omów wybrane elementy składowe systemu automatycznego widzenia..

Efekt W\_04

Podaj przykłady i omów wybrane deskryptory kształtu.

**Forma i warunki zaliczenia:**

Wszystkie zajęcia laboratoryjne (za wyjątkiem pierwszego) są oceniane. W przypadku nieobecności studenta na laboratorium sposób ich odpracowania określa osoba odpowiedzialna za kurs i jest on podany na pierwszych zajęciach.

Na każdym ćwiczeniu prowadzący podaje zakres zadań dla studentów do przygotowania na następne zajęcia (o charakterze praktycznym lub teoretycznym) i wytyczne do ich realizacji. Przygotowanie do zajęć oraz realizacja zadań każdego ćwiczenia są oceniane w skali od 0 do 10 pkt. Łącznie student za 14 zajęć może uzyskać od 0 do 140 pkt..

Warunek uzyskania zaliczenia laboratorium: co najwyżej dwie nieusprawiedliwione nieobecności na laboratoriach i uzyskanie łącznie co najmniej 71 punktów z zajęć (na 140 możliwych)

Zaliczenie laboratorium jest warunkiem koniecznym przystąpienia do egzaminu. Egzamin odbywa się w formie pisemnej, za egzamin można uzyskać maksymalnie 100pkt. Ocena końcowa z zajęć zależy od wyniku laboratorium (w 60%) i egzaminu (w 40%), a końcowy wynik punktowy oblicza się w następujący sposób:

$$P=60(L/140)+40(E/100),$$

gdzie P-końcowy wynik punktowy(maksymalnie 100pkt.) , L-punkty uzyskane z części laboratoryjnej, E-punktowy wynik egzaminu.

Ocena z zajęć zależy od końcowego wyniku punktowego i wyznacza się w następujący sposób.

- 0-50 punktów – 2
- 51-60 punktów – 3
- 61-70 punktów - 3,5
- 71-80 punktów – 4
- 81-90 punktów – 4,5
- 91-100 punktów – 100

Sposób uzyskania punktów:

Laboratorium

1. Ocena udziału w laboratoriach oraz przygotowania się do tych zajęć: 140 pkt. (14 zajęć po 10 pkt.).

Wykład

2. Egzamin pisemny: 100 pkt.

### **Bilans punktów ECTS:**

Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	30 godzin
Udział w ćwiczeniach laboratoryjnych	45 godzin
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	10 godzin

Udział w konsultacjach godz. z przedmiotu	2 godziny
Przygotowanie się do egzaminu i obecność na egzaminie	13 godzin
Sumaryczne obciążenie pracą studenta	100 godzin
Punkty ECTS za przedmiot	<b>4 ECTS</b>
Studia niestacjonarne	
Aktywność	Obciążenie studenta
Udział w wykładach	18 godzin
Udział w ćwiczeniach laboratoryjnych	24 godziny
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	35 godzin
Udział w konsultacjach godz. z przedmiotu	3 godziny
Przygotowanie się do egzaminu i obecność na egzaminie	20 godzin
Sumaryczne obciążenie pracą studenta	100 godzin
Punkty ECTS za przedmiot	<b>4 ECTS</b>

<b>Sylabus przedmiotu / modułu kształcenia</b>		
<b>Nazwa przedmiotu/modułu kształcenia:</b>	<b>Systemy animacji komputerowej</b>	
<b>Nazwa w języku angielskim:</b>	Systems of Computer Animation	
<b>Język wykładowy:</b>	polski	
<b>Kierunek studiów, dla którego przedmiot jest oferowany:</b>	Informatyka	
<b>Jednostka realizująca:</b>	Wydział Nauk Ścisłych i Przyrodniczych	
<b>Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):</b>	obowiązkowy	
<b>Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):</b>	pierwszego stopnia	
<b>Rok studiów:</b>	trzeci	
<b>Semestr:</b>	szósty	
<b>Liczba punktów ECTS:</b>	4	
<b>Imię i nazwisko koordynatora przedmiotu:</b>	dr Mirosław Szaban	
<b>Imię i nazwisko prowadzących zajęcia:</b>	dr Mirosław Szaban	
<b>Założenia i cele przedmiotu:</b>	<p>Celem przedmiotu jest nabycie przez studenta wiedzy na temat technik, standardów oraz dostępnych narzędzi dedykowanych do tworzenia ruchomych obiektów i filmów zarówno w wersji dwuwymiarowej jak i trójwymiarowej. Efektem pozyskanej przez studenta wiedzy jest umiejętność samodzielnego wykonania animacji, filmu ukazującego poruszający się realistyczne obiekt.</p>	
<b>Symbol efektu</b>	<b>Efekt uczenia się: WIEDZA</b>	<b>Symbol efektu kierunkowego</b>
<b>W_01</b>	Ma uporządkowaną wiedzę na temat zagadnień związanych z tworzeniem i wykorzystaniem animacji komputerowej.	<b>K_W01, K_W02, K_W06, K_W08, K_W11, K_W12</b>
<b>W_02</b>	Zna zasady, metody i standardy tworzenia animacji cyfrowych.	<b>K_W01, K_W02, K_W06, K_W08, K_W10, K_W11, K_W12</b>

<b>W_03</b>	Zna wybrane narzędzia i aplikacje służące do tworzenia i obróbki animacji cyfrowych.	<b>K_W01, K_W02, K_W06, K_W08, K_W10, K_W11, K_W12</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: UMIEJĘTNOŚCI</b>	<b>Symbol efektu kierunkowego</b>
<b>U_01</b>	Rozwiązując problemy związane z grafiką komputerową i przetwarzaniem obrazów potrafi dobrać i wykorzystać odpowiednie metody i narzędzia modelowania i programowania, odpowiednie środowiska programistyczne, techniki i technologie programistyczne niezbędne do implementacji animacji komputerowej, oraz potrafi ocenić ich przydatność.	<b>K_U01, K_U02, K_U05, K_U08, K_U10, K_U11</b>
<b>U_02</b>	Potrafi wykonać animację w środowisku Flash.	<b>K_U01, K_U05, K_U08, K_U11, K_U12, K_U13</b>
<b>U_03</b>	Potrafi wykonać animację w środowisku 3 ds. MAX.	<b>K_U01, K_U05, K_U08, K_U11, K_U12, K_U13</b>
<b>U_04</b>	Potrafi wykonać animację poruszającego się człowieka z uwzględnieniem animacji układu kostnego.	<b>K_U01, K_U02, K_U05, K_U08, K_U10, K_U11, K_U12, K_U13, K_U18, K_U19, K_U24</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: KOMPETENCJE SPOŁECZNE</b>	<b>Symbol efektu kierunkowego</b>
<b>K_01</b>	Potrafi współdziałać w grupie przyjmując jej różne role związane z projektowaniem i implementacją systemów grafiki komputerowej.	<b>K_K01, K_K02</b>
<b>K_02</b>	Potrafi wykorzystać metody i techniki charakterystyczne dla systemów grafiki komputerowej do realizacji przedsięwzięć biznesowych.	<b>K_K03</b>
<b>Forma i typy zajęć:</b>	studia stacjonarne: wykłady (30 godz.), ćwiczenia laboratoryjne (45 godz.) studia niestacjonarne: wykłady (18 godz.), ćwiczenia laboratoryjne (24 godz.)	
<b>Wymagania wstępne i dodatkowe:</b>		
1. Umiejętność podstaw programowania. 2. Znajomość zagadnień z zakresu grafiki komputerowej i cyfrowego przetwarzania obrazu i dźwięku.		

## Treści modułu kształcenia:

1. **Animacja poklatkowa.** Korzystanie z linii czasu, Tworzenie klatek kluczowych, Zaznaczanie klatek, Manipulowanie klatkami jednej warstwy, Usuwanie klatek, Tworzenie prostej animacji poklatkowej, Test Movie, Tryb Onion Skin.
2. **Animacje z automatycznym uzupełnianiem klatek.** Płynna animacja ruchu obiektu, Animacja obiektu zmieniającego kolor, Animacja z obracaniem obiektu, Animowanie grafik zmieniających rozmiar, Animacja z metamorfozą, Animacja obiektu po ścieżce, Animacja grafiki z maską.
3. **Skomplikowane zadania animacyjne.** Animacja z obiektem Movie Clip. **Action Script.** Panel Actions. Dołączanie akcji do klatki (Funkcje Movie Control, Goto and stop, Goto and play, Stop, Play), Przypisywanie etykiet do klatki.
4. **Dołączanie dźwięku i wideo.** Importowanie dźwięków, Dołączanie dźwięków do klatek, Import materiału wideo, Dołączanie materiału wideo do klatek. **Interaktywność z obiektami.** Tworzenie prostych przycisków, Tworzenie przycisków zmieniających kształt, Dołączanie akcji do przycisku. **Publikacja filmów w Internecie.** Publikowanie filmów, Eksportowanie filmów Flash do innych formatów.
5. **Animacje zaawansowane.** Animacja w przestrzeni trójwymiarowej (obrót, przemieszczanie obiektów, Global Transform), Narzędzie Deco Tool, Kinematyka odwrotna i kości, Sztuczki i kruczki w animacji Flash. Zwiększanie realizmu animacji: cień i perspektywa.
6. **Importowanie i eksportowanie grafiki oraz multimediiów.** Formaty plików dla obsługiwanych przez technologię Flash dla importu i eksportu, Import i eksport plików Photoshop, Fireworks, Illustrator, Praca z plikami dźwiękowymi, Praca z plikami wideo.
7. **Zaawansowana animacja ruchu.** precyzyjna kontrola i edycja sekwencji animacji przy użyciu nowego narzędzia Motion Editor. Posługiwanie się zestawami ustawień sekwencji animacji. Zagnieżdżanie sekwencji animacji.
8. **Zaawansowane techniki animacji.** Zaawansowane techniki synchronizowania obrazu z dźwiękiem. Wzbogacanie i uatrakcyjnianie animacji elementami i efektami grafiki 3D.
9. **System kości.** Teoria i praktyka animowania obiektów i postaci w sposób interaktywny. Animacja ruchu i kształtu obiektów przy użyciu systemu kości.
10. **Animacja chodu postaci.** technicznie i artystycznie poprawne projektowanie sekwencji animacji kończyn i ciała postaci. Publikowanie animacji w formacie zgodnym ze starszymi wersjami programu Adobe Flash.
11. **Podstawy teoretyczne pracy z grafiką 3D.** Podstawy pracy w pakiecie 3D na przykładzie programu 3ds max, Interfejs programu 3ds MAX, Koncepcja projektu i organizacja sceny 3D, Podstawowe operacje na obiektach.
12. **Modelowanie scen w 3ds MAX I.** Modelowanie w oparciu o wielokąty i powierzchnie podpodziału (subdivision surfaces), Modelowanie w oparciu o powierzchnie NURBS, Modelowanie i kształtowanie obiektów przy użyciu modyfikatorów;
13. **Modelowanie scen w 3ds MAX II.** Tworzenie i edycja map współrzędnych UV, Tworzenie i edycja map tekstur, Oświetlenie i rendering: reguły oświetlania sceny 3D za pomocą różnych źródeł światła (w tym IBL oraz Global Illumination); kontrolowanie cieni; renderowanie obrazów w module wewnętrznym oraz za pomocą modułów Mental Ray i Vray;
14. **Tworzenie animacji w 3ds MAX.** Podstawy tworzenia i animowania postaci w module Character Studio: konstruowanie szkieletu, mięśni i realistycznie odkształcającej się powłoki, Podstawy dynamiki obiektów w scenach statycznych i animacjach: interakcja pomiędzy ciałami sztywnymi, giętkimi i sprężystymi; dynamika tkanin; dynamika włosów, futra i sierści;
15. **Podstawy symulacji zjawisk fizycznych.** Emisja cząsteczek i interakcja cząsteczek z obiektami w scenie, Kompletowanie projektu i prezentowanie zwizualizowanych scen statycznych oraz animacji.

**Literatura podstawowa:**

1. Parent Rick, Animacja komputerowa, Algorytmy i techniki, PWN 2012
2. Tony Parisi, Aplikacje 3D : przewodnik po HTML5, WebGL i CSS3, Helion 2015.
3. Joanna Pasek. 3ds max 2010. Animacja 3D od podstaw. Szkoła efektu. Helion 2010.

**Literatura dodatkowa:**

1. Dev Ramtal, Adrian Dobre. Wprowadzenie do fizyki w grach, animacjach i symulacjach Flash, Helion 2013.
2. Parent R. Computer Animation, Algorithms and Techniques, Morgan Kaufmann, 2002
3. Bill Turner, James Robertson, Richard Bazley. Flash 5. Gry i kreskówki f/x, Helion 2001.

**Planowane formy/działania/metody dydaktyczne:**

Wykład tradycyjny wspomagany technikami multimedialnymi, laboratorium komputerowe wykorzystujące środowiska i aplikacje obróbki obrazu i dźwięku. Zamieszczanie na stronach internetowych problemów, zadań oraz materiałów ćwiczeniowych.

**Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:**

Efekty U\_01 – U\_04 sprawdzane będą na bieżąco, na każdym zajęciach poza pierwszym i ostatnim. Przykładowe zadania:

- Omów rolę animacji układu kostnego w animacji postaci.
- Wymień i krótko scharakteryzuj podstawowe zasady, metody i standardy tworzenia animacji cyfrowych.
- Opisz wybrane narzędzie lub aplikację służące do tworzenia i obróbki animacji cyfrowych.

Efekty W\_01 – W\_03 oraz K\_01 i K\_02 sprawdzane będą na egzaminie pisemnym w sesji egzaminacyjnej. Przykładowe pytania:

- Wykonaj prostą animację układu kostnego postaci, wybierz właściwe narzędzie.
- Wykonaj prostą animację przy pomocy aplikacji Flash, dołącz plik dźwiękowy w tle animacji.
- Korzystając z narzędzi 3ds Max wykonaj animację biegnącego człowieka.

Ponadto efekty U\_01 – U\_04 oraz K\_01 i K\_02 sprawdzane będą poprzez wykonanie zadanie indywidualnego (projektu). Przykładowe zadanie:

- Wykonaj projekt, dobierz właściwe narzędzie i wykonaj animację pojazdu przejeżdżającego przez skrzyżowanie, widok z pierwszej osoby kierowcy pojazdu (kierowca nieruchomy patrzy przed siebie). Kierowca widzi inne pojazdy, ludzi poruszających się chodnikiem oraz przekraczających skrzyżowanie na pasach. Utwórz dokumentację.

**Forma i warunki zaliczenia:**

Moduł kończy się egzaminem. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie laboratorium. Na zaliczenie laboratorium składają się oceny cząstkowe uzyskane na regularnych zajęciach z nauczycielem akademickim oraz z samodzielnie wykonanego zadania indywidualnego według schematu:

Regularne zajęcia – 45 pkt.,

Obrona zadania indywidualnego – 15 pkt.

Zajęcia laboratoryjne będą zaliczone w wypadku uzyskania co najmniej połowy punktów z poszczególnych form aktywności studenta: regularne zajęcia – co najmniej 23 pkt., obrona indywidualnego zadania – co najmniej 8 pkt. Na tej formie zajęć student może maksymalnie uzyskać 60 pkt.

Egzamin jest egzaminem pisemnym. Można na nim uzyskać do 40 pkt. Egzamin będzie zaliczony w przypadku uzyskania co najmniej 20 pkt.

Ocena końcowa z przedmiotu, w zależności od sumy uzyskanych punktów (maksymalnie 100 pkt) jest następująca (w nawiasach ocena wg skali ECTS):

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

#### **Bilans punktów ECTS:**

##### Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	30 godz.
Udział w ćwiczeniach laboratoryjnych	45 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	10 godz.
Udział w konsultacjach godz. z przedmiotu	5 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	10 godz.
<b>Sumaryczne obciążenie pracą studenta</b>	<b>100 godz.</b>
<b>Punkty ECTS za przedmiot</b>	<b>4 ECTS</b>

##### Studia niestacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	18 godz.
Udział w ćwiczeniach laboratoryjnych	24 godz.



Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	24 godz.
Udział w konsultacjach godz. z przedmiotu	10 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	24 godz.
<b>Sumaryczne obciążenie pracą studenta</b>	<b>100 godz.</b>
<b>Punkty ECTS za przedmiot</b>	<b>4 ECTS</b>

<b>Sylabus przedmiotu / modułu kształcenia</b>		
<b>Nazwa przedmiotu/modułu kształcenia:</b>	<b>Silniki gier komputerowych</b>	
<b>Nazwa w języku angielskim:</b>	Computer game engines	
<b>Język wykładowy:</b>	polski	
<b>Kierunek studiów, dla którego przedmiot jest oferowany:</b>	Informatyka	
<b>Jednostka realizująca:</b>	Wydział Nauk Ścisłych i Przyrodniczych	
<b>Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):</b>	fakultatywny	
<b>Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):</b>	pierwszego stopnia	
<b>Rok studiów:</b>	trzeci	
<b>Semestr:</b>	szósty	
<b>Liczba punktów ECTS:</b>	3	
<b>Imię i nazwisko koordynatora przedmiotu:</b>	dr Andrzej Salamończyk	
<b>Imię i nazwisko prowadzących zajęcia:</b>	dr Andrzej Salamończyk	
<b>Założenia i cele przedmiotu:</b>	Celem kursu jest zapoznanie studentów z zasadami działania silników gier komputerowych oraz z ich przykładowymi implementacjami	
<b>Symbol efektu</b>	<b>Efekt uczenia się: WIEDZA</b>	<b>Symbol efektu kierunkowego</b>
<b>W_01</b>	Zna i rozumie zasadę działania silników graficznych	<b>K_W08</b>
<b>W_02</b>	Zna i rozumie architekturę silników graficznych	<b>K_W08</b>
<b>W_03</b>	Zna i rozumie zastosowania silników graficznych	<b>K_W08</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: UMIEJĘTNOŚCI</b>	<b>Symbol efektu kierunkowego</b>
<b>U_01</b>	Potrafi utworzyć grę korzystając z silnika Unity 3D	<b>K_U011</b>
<b>U_02</b>	Potrafi utworzyć teren i dodać obiekty (assets) oraz postacie do gry w Unity 3D	<b>K_U11, K_U17, K_U19</b>
<b>U_03</b>	Potrafi dodać systemy cząsteczkowe oraz uwzględniać prawa fizyki w Unity 3D	<b>K_U11, K_U17, K_U19</b>

<b>U_04</b>	Potrafi dodawać do sceny oświetlenia i dźwięki w Unity3D.	<b>K_U11, K_U17, K_U19</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: KOMPETENCJE SPOŁECZNE</b>	<b>Symbol efektu kierunkowego</b>
<b>K_01</b>	Jest gotów do podejmowania decyzji dotyczącej wyboru odpowiednich rozwiązań i narzędzi oraz krytycznej oceny własnych rozwiązań	<b>K_K01</b>
<b>K_02</b>	Jest gotów do uznania znaczenia wiedzy dotyczącej silników gier komputerowych oraz krytycznie potrafi ocenić swoje działania	<b>K_K01</b>
<b>Forma i typy zajęć:</b>	studia stacjonarne: wykłady (30 godz.), ćwiczenia laboratoryjne (30 godz.) studia niestacjonarne: wykłady (15 godz.), ćwiczenia laboratoryjne (15 godz.)	
<b>Wymagania wstępne i dodatkowe:</b>		
<ol style="list-style-type: none"> <li>1. Znajomość podstawowych pojęć i algorytmów grafiki komputerowej.</li> <li>2. Umiejętność programowania w języku obiektowym (C#).</li> </ol>		
<b>Treści modułu kształcenia:</b>		
<p>Wykłady:</p> <ol style="list-style-type: none"> <li>16. Architektura silników gier 3D</li> <li>17. Rozwój silników gier komputerowych</li> <li>18. Silnik Unity</li> <li>19. Zarządzanie grafem sceny w silnikach gier.</li> <li>20. Zarządzanie postaciami w silnikach gier. Interakcje. Kolidzje. Animacje.</li> <li>21. Renderer w silnikach gier.</li> <li>22. Postprocesing. Systemy cząsteczkowe w silnikach gier. Oświetlenie.</li> <li>23. Fizyka w silnikach gier. Audio w silnikach gier.</li> <li>24. Sztuczna inteligencja w silnikach gier.</li> <li>25. Profilowanie i debugowanie w silnikach gier.</li> <li>26. Silnik UDK, Unreal Engine.</li> <li>27. Silniki OGRE.</li> <li>28. Silnik Irrlicht.</li> <li>29. Przegląd i porównanie innych silników gier komputerowych.</li> <li>30. Zastosowania silników graficznych. Inne technologie wspomagające tworzenie gier.</li> </ol> <p>Laboratoria:</p> <ol style="list-style-type: none"> <li>1. Środowisko Unity 3D</li> <li>2. Zaprojektowanie scenariusza gry.</li> <li>3. Przygotowywanie terenu. Prefabrykaty.</li> <li>4. Dodawanie obiektów do sceny. Modele, materiały i tekstury.</li> <li>5. Dodawanie postaci do sceny i sterowanie nimi.</li> <li>6. Obsługa zdarzeń i kolizji.</li> </ol>		

7. Tworzenie animacji.
8. Modelowanie oświetlenia. Korzystanie z kamery.
9. Dodawanie elementów fizyki do gry. Dodawanie dźwięku.
10. Dodawanie efektów z użyciem systemów cząsteczkowych.
11. Dodawanie elementów sztucznej inteligencji i profilowania postaci.
12. Testowanie i debugowanie gry
13. Udoskonalanie gry.
14. Przygotowanie dokumentacji.
15. Przegląd i prezentacja utworzonych gier

#### **Literatura podstawowa:**

9. Mike Geig. Unity. Przewodnik projektanta gier.
10. Jeremy Gibson Bond. Projektowanie gier przy użyciu środowiska Unity i języka C#. Od pomysłu do gotowej gry. Wydanie II. Helion 2018.

#### **Literatura dodatkowa:**

1. Joanna Lee. Unreal Engine. Nauka pisania gier dla kreatywnych
2. Jason Gregory. Game Engine Architecture, Third Edition. A K Peters/CRC Press, 2018

#### **Planowane formy/działania/metody dydaktyczne:**

Wykład tradycyjny wspomagany technikami multimedialnymi. Zamieszczanie na stronach internetowych zadań i materiałów do laboratoriów.

#### **Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:**

Efekty U\_01 – U\_04 są sprawdzane w czasie ocenianych zadań na laboratoriach.

Efekty W\_01 – W\_03, K\_01, K\_02 sprawdzane są egzaminie.

#### **Forma i warunki zaliczenia:**

Wszystkie zajęcia laboratoryjne (za wyjątkiem pierwszego) są oceniane. W przypadku nieobecności studenta na laboratorium sposób ich odpracowania określa osoba odpowiedzialna za kurs i jest on podany na pierwszych zajęciach.

Na każdym ćwiczeniu prowadzący podaje zakres zadań dla studentów do przygotowania na następne zajęcia (o charakterze praktycznym lub teoretycznym) i wytyczne do ich realizacji. Przygotowanie do zajęć oraz realizacja zadań każdego ćwiczenia są oceniane w skali od 0 do 10 pkt. Łącznie student za 14 zajęć może uzyskać od 0 do 140 pkt..

Warunek uzyskania zaliczenia laboratorium: co najwyżej dwie nieusprawiedliwione nieobecności na laboratoriach i uzyskanie łącznie co najmniej 71 punktów z zajęć (na 140 możliwych)

Zaliczenie laboratorium jest warunkiem koniecznym przystąpienia do egzaminu. Egzamin odbywa się w formie pisemnej, za egzamin można uzyskać maksymalnie 100pkt. Ocena końcowa z zajęć zależy od wyniku laboratorium (w 60%) i egzaminu (w 40%), a końcowy wynik punktowy oblicza się w następujący sposób:

$$P=60(L/140)+40(E/100),$$

gdzie P-końcowy wynik punktowy(maksymalnie 100pkt.) , L-punkty uzyskane z części laboratoryjnej, E-punktowy wynik egzaminu.

Ocena z zajęć zależy od końcowego wyniku punktowego i wyznacza się w następujący sposób.

- 0-50 punktów – 2
- 51-60 punktów – 3
- 61-70 punktów - 3,5
- 71-80 punktów – 4
- 81-90 punktów – 4,5
- 91-100 punktów – 100

Sposób uzyskania punktów:

Laboratorium

1. Ocena udziału w laboratoriach oraz przygotowania się do tych zajęć: 140 pkt. (14 zajęć po 10 pkt.).

Wykład

2. Egzamin pisemny: 100 pkt.

### **Bilans punktów ECTS:**

#### Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	30 godzin
Udział w ćwiczeniach laboratoryjnych	30 godzin
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	8 godzin
Udział w konsultacjach godz. z przedmiotu	2 godziny
Przygotowanie się do egzaminu i obecność na egzaminie	5 godzin
Sumaryczne obciążenie pracą studenta	75 godzin
Punkty ECTS za przedmiot	<b>3 ECTS</b>

#### Studia niestacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	15 godzin
Udział w ćwiczeniach laboratoryjnych	15 godzin

Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	28 godzin
Udział w konsultacjach godz. z przedmiotu	2 godziny
Przygotowanie się do egzaminu i obecność na egzaminie	15 godzin
Sumaryczne obciążenie pracą studenta	75 godzin
Punkty ECTS za przedmiot	<b>3 ECTS</b>

**Specjalność: Mobilne systemy komputerowe, semestr VI**

<b>Sylabus przedmiotu / modułu kształcenia</b>		
<b>Nazwa przedmiotu/modułu kształcenia:</b>		<b>Programowanie systemów mobilnych</b>
<b>Nazwa w języku angielskim:</b>		Mobile Systems Programming
<b>Język wykładowy:</b>	polski	
<b>Kierunek studiów, dla którego przedmiot jest oferowany:</b>		Informatyka
<b>Jednostka realizująca:</b>		Wydział Nauk Ścisłych i Przyrodniczych
<b>Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):</b>		fakultatywny
<b>Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):</b>		pierwszego stopnia
<b>Rok studiów:</b>	trzeci	
<b>Semestr:</b>	szósty	
<b>Liczba punktów ECTS:</b>	5	
<b>Imię i nazwisko koordynatora przedmiotu:</b>		dr Mirosław Szaban
<b>Imię i nazwisko prowadzących zajęcia:</b>		dr Mirosław Szaban
<b>Założenia i cele przedmiotu:</b>		Celem wykładu jest zapoznanie studentów: historią i architekturą systemu Android (w tym z modelem bezpieczeństwa tego systemu). Student pozna praktyki i wzorce do zaawansowanego tworzenia oprogramowania w systemie Androida oraz przykłady usług zewnętrznych. Ponadto, na wykładach zostanie omówiony i porównany (z Androidem) system IOS, Windows Phone, inne np., BlackBerry OS.
<b>Symbol efektu</b>	<b>Efekt uczenia się: WIEDZA</b>	<b>Symbol efektu kierunkowego</b>
<b>W_01</b>	Student zna zasady projektowania i tworzenia poprawnych aplikacji na Android.	<b>K_W02, K_W03, K_W05, K_W07</b>
<b>W_02</b>	Zna zasady komunikacji wewnętrznej i międzyprocesowej w systemie Android.	<b>K_W02, K_W03, K_W05, K_W07</b>
<b>W_03</b>	Zna zasady tworzenia: graficznego interfejsu użytkownika dla aplikacji mobilnych z systemem Android, korzystania z aplikacji mobilnych z multimediami, aplikacji sieciowych i połączeń telefonicznych, pobierania i utrwalania danych.	<b>K_W02, K_W03, K_W05, K_W07, K_W09</b>



<b>W_04</b>	Zna zasady tworzenia gier i animacji na urządzenia mobilne w systemie Android.	<b>K_W02, K_W03, K_W05, K_W07</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: UMIEJĘTNOŚCI</b>	<b>Symbol efektu kierunkowego</b>
<b>U_01</b>	Student potrafi utworzyć (zaprojektować i zaimplementować) przykładową aplikację mobilną na urządzenie Android, korzystając z platformy Android Studio.	<b>K_U01, K_U06, K_U10, K_U11, K_U12, K_U15, K_U24</b>
<b>U_02</b>	Umie stworzyć mobilną aplikację wyświetlającą aktualną pozycję użytkownika, wykorzystującą lokalizację GPS i mapy.	<b>K_U01, K_U06, K_U10, K_U11, K_U12, K_U15, K_U24</b>
<b>U_03</b>	Potrafi wykonać aplikację wizualizującą wektorowe dane przestrzenne pochodzące z sensorów lub pobierane z bazy danych.	<b>K_U01, K_U06, K_U10, K_U11, K_U12, K_U15, K_U24</b>
<b>U_04</b>	Potrafi utworzyć aplikacje wykorzystujące czujniki m.in. akcelerometr, połączenie urządzenia z Bluetoothem.	<b>K_U01, K_U06, K_U10, K_U11, K_U12, K_U15, K_U24</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: KOMPETENCJE SPOŁECZNE</b>	<b>Symbol efektu kierunkowego</b>
<b>K_01</b>	Potrafi formułować opinie na temat zagadnień z zakresu systemów oraz sieci mobilnych i bezprzewodowych, między innymi nt. jakości rozwiązań stosowanych w ich koncepcji i budowie.	<b>K_K01</b>
<b>Forma i typy zajęć:</b>	studia stacjonarne: wykłady (30 godz.), ćwiczenia laboratoryjne (45 godz.) studia niestacjonarne: wykłady (18 godz.), ćwiczenia laboratoryjne (24 godz.)	
<b>Wymagania wstępne i dodatkowe:</b>		
1. Umiejętność programowania w C++/Java 2. Technologie sieciowe 3. Technologie mobilne		
<b>Treści modułu kształcenia:</b>		
<b>Wykłady:</b>		
1. <b>Wprowadzenie – znaczenie systemów mobilnych.</b> Zapoznanie z ideą przetwarzania mobilnego. Pokazanie rozkwitu dziedziny oraz czynników wpływających na jej rozwój. Przedstawienie mnogości		

zastosowań, olbrzymich korzyści dla klienta końcowego oraz złożonych i nietrywialnych problemów, jakie stoją przed projektantami nowoczesnych systemów mobilnych.

2. Porównanie systemów operacyjnych dla urządzeń mobilnych z Androidem, system IOS, Windows Mobile (Phone), BlackBerry OS, inne ...
3. **Przegląd i porównanie popularnych mobilnych systemów operacyjnych.** Wprowadzenie do architektury aplikacji tworzonych dla różnych systemów operacyjnych, omówienie narzędzi i środowisk programistycznych.
4. Programowanie aplikacji dla systemu Android, oraz dla systemu Windows mobile w języku Java.
5. Podstawy wielowątkowości w Javie i systemie Android.
6. Komunikacja wątków w obrębie procesu i pomiędzy procesami.
7. Zastosowanie strategii redukcji ryzyka powstawania wycieków pamięci.
8. Zarządzanie cyklem życia wątku podstawowego.
9. Sekwencyjne uruchamianie zadań w wątku tła za pomocą klasy HandlerThread
10. Używanie frameworku wykonawcy Javy do kontrolowania i anulowania wątków.
11. Obsługa wykonywania zadań w tle za pomocą klas AsyncTask oraz IntentService.
12. Uzyskiwanie dostępu do dostawców treści za pomocą klasy AsyncQueryHandler.
13. Używanie ładowarek do aktualizowania interfejsu użytkownika nowymi danymi.
14. Najnowsza wiedza z zakresu przedmiotu. Współczesne technologie mobilne i bezprzewodowe.
15. Biznes w świecie mobile : jak zaprojektować, wykonać i wypromować aplikację mobilną.

#### Laboratoria:

1. **Wprowadzenie do Androida - przypomnienie podstaw.** Android Studio: narzędzia, interfejs użytkownika, funkcje i pluginy, narzędzia do analizy aplikacji, Gradle,
2. **Struktura aplikacji:** cykl życia aktywności, zachowanie stanu aktywności, elementy interfejsu użytkownika.
3. **Tworzenie aplikacji:** zasoby aplikacji, podstawowe elementy interfejsu, menu opcji, interfejs, 4.6. Używanie widoków za pomocą Butter Knife, komunikacja z użytkownikiem.
4. **Grupowanie widoków:** podstawowe layouty (LinearLayout i RelativeLayout), ConstraintLayout, widok listy – ListView, własny adapter listy, widok listy – RecyclerView.
5. **Aplikacja z wieloma ekranami – aktywnościami:** komunikacja między aktywnościami – Intent, przekazywanie danych do innych aktywności, nawigacja w aplikacji, aplikacja z wieloma ekranami – fragmenty, zastosowanie fragmentów.
6. **Wielowątkowość w Androidzie (przetwarzanie asynchroniczne):** modyfikacja interfejsu użytkownika przez wątki, ograniczenia - Application Not Responding, użycie uchwytów do komunikacji, AsyncTask, Loadery, Service i zastosowanie, biblioteki RxJava.
7. **Architektura aplikacji:** wprowadzenie do Clean Architecture i MVP, biblioteki, komunikacja sieciowa, HTTP Networking, format JSON, Biblioteka Retrofit.
8. **Data Storage:** wprowadzenie do SQLite, zastosowanie SQLite, ContentProvidery, alternatywy: Realm, Firebase.
9. **Komunikacja sieciowa:** obsługa sieci, Web service typu RESTFul, bezpieczeństwo danych w sieci, użycie protokołów sieciowych w asynchronicznych aplikacjach Androida.
10. **Lokalizacja i sensory:** Lokalizacja (GPS), kontrola uprawnień, sensory w Androidzie.
11. **Android Architecture Components:** Wprowadzenie do Android Architecture Components, Room, Obsługa cyklu życia.

12. **Programowanie modułowe na platformie Android:** paradygmat programowania modułowego, zastosowanie wzorców projektowych do programowania w języku Java, dynamiczne wstrzykiwanie zależności, dobre praktyki i wzorce projektowe charakterystyczne dla systemu Android, zaawansowane tworzenie interfejsu użytkownika.
13. **Testowanie aplikacji Androida:** narzędzie służące do testowania aplikacji w kontekście systemu Android, testowanie jednostkowe z użyciem bibliotek Robolectric i RoboSpock, tworzenie testów funkcjonalnych aplikacji.
14. **Publikacja aplikacji:** zaciemnianie kodu aplikacji – ProGuard, optymalizacja rozmiaru pliku APK, monitoring życia aplikacji, przygotowanie do publikacji, Google Play Developer Console.
15. **Usługi Google dla aplikacji Androida:** Google Maps, tworzenie kopii zapasowych danych użytkownika, zapobieganie nielegalnej dystrybucji oprogramowania.

#### Literatura podstawowa:

1. Android Studio : wygodne i efektywne tworzenie aplikacji / Adam Gerber, Clifton Craig ; [tłumaczenie Rafał Jońca]. - Gliwice : Helion, cop. 2016. - ISBN 978-83-283-2009-3
2. Anders Gransson, Android. Aplikacje wielowątkowe. Techniki przetwarzania. O'Reilly, 2015.
3. Ian F. Darwin, Android. Receptury, 3rd Edition, Helion, 2015
4. Biznes w świecie mobile : jak zaprojektować, wykonać i wypromować aplikację mobilną / Sylwia Żółkiewska, Małgorzata Rycharska, Noemi Gryczko. - Warszawa : Poltext, 2018. - ISBN 978-83-7561-866-2

#### Literatura dodatkowa:

1. Wydajne aplikacje dla systemu Android : programuj szybko i efektywnie / Doug Sillars ; [tłumaczenie: Andrzej Watrak]. - Gliwice : Wydawnictwo Helion, cop. 2017. - ISBN 978-83-283-2935-5
2. Michał Szczepanik, Android. Poziom drugi. Zaawansowane programowanie aplikacji. Kurs video, 2017.
3. Google, <http://developer.android.com/index.html>

#### Planowane formy/działania/metody dydaktyczne:

Wykład tradycyjny wspomagany technikami multimedialnymi, ćwiczenia wspomagane technikami multimedialnymi oraz środowiskami umożliwiającymi programowanie i symulację systemów mobilnych. Zamieszczanie na stronach internetowych problemów i zadań ćwiczeniowych.

#### Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:

Efekty W\_01 – W\_04 sprawdzane będą na egzaminie pisemnym w sesji egzaminacyjnej, jako zagadnienia teoretyczne. Przed egzaminem studenci będą mieli dostęp do pełnej listy pytań. Przykładowe pytania:

- Omów architekturę systemu Android.
- Scharakteryzuj System Mobilny. Wymień jego elementy i krótko opisz.
- Wymień znane środowiska programistyczne aplikacji mobilnych, scharakteryzuj wybrane.

Efekty U\_01 – U\_04 sprawdzane będą na bieżąco, na każdych zajęciach poza pierwszym i ostatnim w postaci zadań praktycznych. Tematyka następnego laboratorium będzie podana tydzień przed zajęciami.

Student, na podstawie podanej literatury, musi się do nich przygotować samodzielnie lub korzystając z konsultacji. Przykładowe zadanie:

Utworzyć aplikację mobilną, która korzystając z map oraz modułu GPS, wskaże na mapie aktualną pozycję.

Ponadto efekty U\_01 oraz efekt K\_01 będzie sprawdzany poprzez wykonanie zadania indywidualnego – projektu implementacyjnego aplikacji mobilnej. Tematy zadań indywidualnych zostaną przedstawione najpóźniej na 3 zajęciach.

#### **Forma i warunki zaliczenia:**

Moduł kończy się egzaminem. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie laboratorium. Na zaliczenie laboratorium składają się oceny cząstkowe uzyskane na regularnych zajęciach z nauczycielem akademickim, za które można uzyskać maksymalnie 40 pkt. oraz ocena za wykonanie zadania indywidualnego, za które można otrzymać 20 pkt.. Maksymalna liczba punktów na tej formie zajęć wynosi 60 pkt. Zaliczenie zajęć laboratoryjnych i dopuszczenie do egzaminu jest możliwe po uzyskaniu co najmniej 30 pkt., gdzie co najmniej 20 pkt stanowi ocena zajęć laboratoryjnych oraz 10 pkt. ocena zadania indywidualnego.

Egzamin jest egzaminem pisemnym. Można na nim uzyskać do 40 pkt. Egzamin będzie zaliczony w przypadku uzyskania co najmniej 20 pkt.

Ocena końcowa z przedmiotu, w zależności od sumy uzyskanych punktów (maksymalnie 100 pkt) jest następująca (w nawiasach ocena wg skali ECTS):

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

#### **Bilans punktów ECTS:**

Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	30 godz.
Udział w ćwiczeniach laboratoryjnych	45 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	20 godz.
Udział w konsultacjach godz. z przedmiotu	10 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	20 godz.

<b>Sumaryczne obciążenie pracą studenta</b>	<b>125 godz.</b>
<b>Punkty ECTS za przedmiot</b>	<b>5 ECTS</b>
Studia niestacjonarne	
Aktywność	Obciążenie studenta
Udział w wykładach	18 godz.
Udział w ćwiczeniach laboratoryjnych	24 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	41 godz.
Udział w konsultacjach godz. z przedmiotu	10 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	32 godz.
<b>Sumaryczne obciążenie pracą studenta</b>	<b>125godz.</b>
<b>Punkty ECTS za przedmiot</b>	<b>5 ECTS</b>

<b>Sylabus przedmiotu / modułu kształcenia</b>		
<b>Nazwa przedmiotu/modułu kształcenia:</b>	<b>Mobilne systemy operacyjne</b>	
<b>Nazwa w języku angielskim:</b>	Mobile operating systems	
<b>Język wykładowy:</b>	polski	
<b>Kierunek studiów, dla którego przedmiot jest oferowany:</b>	informatyka	
<b>Jednostka realizująca:</b>	Wydział Nauk Ścisłych i Przyrodniczych	
<b>Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):</b>	fakultatywny	
<b>Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):</b>	pierwszego stopnia	
<b>Rok studiów:</b>	trzeci	
<b>Semestr:</b>	szósty	
<b>Liczba punktów ECTS:</b>	4	
<b>Imię i nazwisko koordynatora przedmiotu:</b>	dr Piotr Świtalski	
<b>Imię i nazwisko prowadzących zajęcia:</b>	dr Piotr Świtalski	
<b>Założenia i cele przedmiotu:</b>	<p>Celem przedmiotu jest zaznajomienie studentów z problematyką mobilnych systemów operacyjnych w stopniu wystarczającym do zrozumienia tej tematyki. Przedmiot ma za zadanie pogłębić wiedzę o systemach operacyjnych. Wprowadza w tematykę maszyn wirtualnych i zarządzania ograniczonymi zasobami w mobilnych systemach operacyjnych. W założeniach do tego przedmiotu przewiduje się zajęcia praktyczne z użyciem komputerów, podczas których studenci nabędą umiejętności programowania w mobilnych systemach operacyjnych.</p>	
<b>Symbol efektu</b>	<b>Efekt uczenia się: WIEDZA</b>	<b>Symbol efektu kierunkowego</b>
<b>W_01</b>	Zna i rozumie zagadnienia z zakresu działania mobilnych systemów operacyjnych.	<b>K_W05, K_W12, K_W15</b>
<b>W_02</b>	Zna i rozumie mechanizmy zarządzania procesami oraz pamięcią operacyjną związane z mobilnymi systemami operacyjnymi	<b>K_W05, K_W15</b>

<b>W_03</b>	Zna i rozumie zagadnienia związane z obecnymi rozwiązaniami wprowadzonych w systemach mobilnych	<b>K_W10, K_W15</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: UMIEJĘTNOŚCI</b>	<b>Symbol efektu kierunkowego</b>
<b>U_01</b>	Potrafi zaimplementować interpreter kodu dla maszyny wirtualnej	<b>K_U11, K_U13, K_U18, K_U26</b>
<b>U_02</b>	Potrafi ocenić przyjęte rozwiązania w stosunku do wydajności działania systemów mobilnych	<b>K_U17, K_U26</b>
<b>U_03</b>	Potrafi ocenić stopień bezpieczeństwa w mobilnym systemie operacyjnym	<b>K_U17, K_U26</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: KOMPETENCJE SPOŁECZNE</b>	<b>Symbol efektu kierunkowego</b>
<b>K_01</b>	Jest gotów do podejmowania krytycznych decyzji związanych z projektowaniem i implementacją mechanizmów mobilnych systemów operacyjnych, jest gotów do krytycznej oceny własnych działań oraz do konstruktywnej krytyki	<b>K_K01, K_K04</b>
<b>Forma i typy zajęć:</b>	Studia stacjonarne: wykłady (30 godz.), ćwiczenia laboratoryjne (45 godz.) Studia niestacjonarne: wykłady (18 godz.), ćwiczenia laboratoryjne (24 godz.)	
<b>Wymagania wstępne i dodatkowe:</b>		
Warunkiem uczestnictwa w zajęciach jest znajomość zagadnień z następujących przedmiotów: „Systemy operacyjne”, „Systemy wbudowane”, „Programowanie obiektowe” lub znajomość literatury obowiązującej w tych przedmiotach.		
<b>Treści modułu kształcenia:</b>		
<ol style="list-style-type: none"> <li><b>Wprowadzenie do mobilnych systemów operacyjnych (SO).</b> Charakterystyka systemów mobilnych. Porównanie systemów operacyjnych przeznaczonych na urządzenia mobilne. Rozwój mobilnych systemów operacyjnych. Założenia projektowe dla takich systemów.</li> <li><b>Budowa i architektura procesorów przeznaczonych do urządzeń mobilnych.</b> Architektury procesorów ARM oraz bazujących na rodzinie procesorów x86. Systemy wbudowane. Układy „System On Chip”. Układy ASIC oraz FPGA. Procesory wielordzeniowe. Zarządzanie energią w procesorach.</li> <li><b>Mikrojądro systemu operacyjnego.</b> Architektura mikrojądra. Rodzaje mikrojąder. Jądro monolityczne. Jądro hybrydowe. Przykłady jąder dla SO. Własności jądra SO przeznaczonego na urządzenia mobilne.</li> <li><b>Wprowadzenie do systemu Android.</b> Zarządzanie pamięcią. Mechanizm RPC/IPC. Podsystem dziennika zdarzeń. Budowa systemu Android. Zarządzanie modułami sprzętowymi urządzenia mobilnego. System plików. Biblioteki systemu Android. Demony. Narzędzia systemowe. Narzędzie ADB.</li> </ol>		

5. **Wprowadzenie do systemu iOS.** Jądro systemu iOS. Zarządzanie pamięcią. Zarządzanie modułami sprzętowymi urządzenia mobilnego. Podsystem zasilania. Struktura systemu plików.
6. **Procesy i wątki.** Tworzenie procesu. Zarządzanie procesami w systemach mobilnych. Tworzenie wątków i ich obsługa. Programowanie z użyciem wątków.
7. **Maszyny wirtualne.** Budowa maszyny wirtualnej. Interpretery kodu. Maszyny wirtualne bazujące na jądrze. Wirtualizacja sprzętowa. Wirtualizacja programowa. Wirtualizacja na poziomie systemu operacyjnego. Maszyny wirtualne dla urządzeń mobilnych.
8. **Maszyny wirtualne urządzeń mobilnych cz. 1.** Koncepcja maszyny wirtualnej dla urządzeń mobilnych. Architektura maszyny wirtualnej.
9. **Maszyny wirtualne urządzeń mobilnych cz. 2.** Zarządzanie usługami. Uruchomienie (boot sequence) maszyny. Środowisko uruchomieniowe Android Runtime (ART).
10. **Maszyna wirtualna Javy (JVM).** Specyfikacja maszyny JVM. Budowa maszyny JVM. Class loader. Wprowadzenie do kodu bajtowego Javy. Języki kompilowane do kodu bajtowego Javy. Sprzętowa implementacja maszyny Javy.
11. **Kod bajtowy Java cz.1.** Budowa pliku .class. Instrukcje i operacje. Konwersja typów. Wywołanie metod. Generowanie wyjątków.
12. **Kod bajtowy Java cz.2.** Generowanie kodu bajtowego Javy. Uruchomienie kodu bajtowego Javy. Weryfikacja kodu.
13. **Zarządzanie energią w mobilnych systemach operacyjnych.** Sposoby redukcji energii wykorzystywanej przez mobilne procesory. Szeregowanie zadań w systemach mobilnych. Zarządzanie energią przez maszynę wirtualną oraz ART.
14. **Bezpieczeństwo mobilnego systemu operacyjnego.** Zarządzanie i hermetyzacja procesów. Zarządzanie uprawnieniami w SELinux. Ochrona systemu plików. Bezpieczeństwo jądra SO. Bezpieczeństwo aplikacji SO Android.
15. **Sieci sensorowe.** Architektura sieci sensorowych. Samoorganizacja węzłów sieci sensorowych. Zarządzanie energią. Algorytmy wykorzystywane w sieciach sensorowych.

#### Literatura podstawowa:

1. Elenkov N., Android Security Internals: An In-Depth Guide to Android's Security Architecture 1st Edition, No Starch Press 2014,
2. The Java Virtual Machine Specification Java SE 15 Edition.

#### Literatura dodatkowa:

1. Chong C.: Kumar S. P.: Sensor Networks: Evolution Opportunities and Challenges, Proceedings of the IEEE, 2003.
2. Naghdy F., Bu S.: Wireless Ad-hoc Networks, 3rd IEEE International Conference on Industrial Informatics (INDIN), 2005.

#### Planowane formy/działania/metody dydaktyczne:

Wykład tradycyjny wspomagany jest technikami multimedialnymi. Ćwiczenia laboratoryjne – zajęcia praktyczne z wykorzystaniem wybranych narzędzi programowych. Na stronie internetowej prowadzącego zamieszczane są materiały z problemami i zadaniami laboratoryjnymi.



### Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:

Efekty W\_01 do W\_03 zostaną zweryfikowane podczas egzaminu pisemnego w trakcie trwania sesji egzaminacyjnej. Na egzaminie pytania będą dotyczyły poznanych zagadnień związanych z mobilnymi systemami operacyjnymi. Przykładowe pytania egzaminacyjne:

- Przedstaw budowę maszyny wirtualnej.
- Przedstaw architekturę systemu Android.

Przed egzaminem studenci będą mieli dostęp do przykładowych pytań.

Efekty U\_01 do U\_03 będą sprawdzane systematycznie na zajęciach laboratoryjnych. Efekt K\_01 weryfikowany w oparciu o posiadaną wiedzę i umiejętności w czasie zajęć laboratoryjnych oraz podczas realizacji projektu indywidualnego.

Przykładowe zadania:

- Opracuj uproszczony interpreter kodu dla maszyny wirtualnej.
- Zaprojektuj aplikację mobilną oferującą przedstawioną w dalszej części zadania funkcjonalność.

Materiały na następne laboratorium będą dostępne na dwa dni przed zajęciami.

### Forma i warunki zaliczenia:

Ocena z przedmiotu składa się z trzech ocen cząstkowych:

- oceny z zajęć laboratoryjnych,
- oceny z projektu indywidualnego,
- oceny z egzaminu końcowego.

Na ocenę z zajęć laboratoryjnych składają się oceny cząstkowe uzyskane na regularnych zajęciach z nauczycielem akademickim, za które można uzyskać sumarycznie 24 pkt. Dodatkowo student jest zobowiązany do realizacji projektu indywidualnego, z którego może otrzymać maksymalnie 16 pkt. Zaliczenie zajęć laboratoryjnych możliwe jest po pozytywnie zakończonej obronie projektu indywidualnego i uzyskaniu sumarycznie co najmniej 51% liczby punktów z tej formy zaliczenia.

W trakcie sesji odbędzie się egzamin końcowy. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie laboratorium. Egzamin przewidziany jest w formie pisemnej. Można na nim uzyskać maksymalnie 60 pkt. Egzamin będzie zaliczony w przypadku uzyskania co najmniej 51% liczby punktów z tej formy zaliczenia. Ocena końcowa z przedmiotu, w zależności od sumy uzyskanych punktów (maksymalnie 100 pkt) jest następująca (w nawiasach ocena wg skali ECTS):

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

### Bilans punktów ECTS:

Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	30 godz.
Udział w ćwiczeniach laboratoryjnych	45 godz.
Przygotowanie projektu indywidualnego	10 godz.
Przygotowanie się do egzaminu	8 godz.
Przygotowanie się do ćwiczeń laboratoryjnych	5 godz.
Udział w konsultacjach z przedmiotu	2 godz.
<b>Sumaryczne obciążenie pracą studenta</b>	<b>100 godz.</b>
<b>Punkty ECTS za przedmiot</b>	<b>4 ECTS</b>
Studia niestacjonarne	
Aktywność	Obciążenie studenta
Udział w wykładach	18 godz.
Udział w ćwiczeniach laboratoryjnych	24 godz.
Przygotowanie projektu indywidualnego	10 godz.
Przygotowanie się do egzaminu	22 godz.
Przygotowanie się do ćwiczeń laboratoryjnych	24 godz.
Udział w konsultacjach z przedmiotu	2 godz.
<b>Sumaryczne obciążenie pracą studenta</b>	<b>100 godz.</b>
<b>Punkty ECTS za przedmiot</b>	<b>4 ECTS</b>

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:		Środowiska programowania systemów robotycznych
Nazwa w języku angielskim:		Programming environments robotic systems
Język wykładowy:	polski	
Kierunek studiów, dla którego przedmiot jest oferowany:		informatyka
Jednostka realizująca:	Wydział Nauk Ścisłych i Przyrodniczych	
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):		fakultatywny
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):		pierwszego stopnia
Rok studiów:	trzeci	
Semestr:	szósty	
Liczba punktów ECTS:	4	
Imię i nazwisko koordynatora przedmiotu:		dr Grzegorz Terlikowski
Imię i nazwisko prowadzących zajęcia:		dr Grzegorz Terlikowski
Założenia i cele przedmiotu:		<p>Celem kursu jest zapoznanie studentów w podstawowe zagadnienia związane z robotami (pojęcia, zastosowania), z paradygmatami architektonicznymi budowy robota, w tym z robotami mobilnymi oraz softwarowego.</p> <p>W szczególności celem kursu jest nauczenie studentów pracy w wybranych środowiskach robotycznych.</p>
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	Student zna i rozumie podstawowe pojęć w zakresie robotyki,	K_W05. K_W06, K_W07
W_02	Zna i rozumie podstawowe zastosowania robotów i rozumie rosnący ich udział w życiu człowieka,	K_W05. K_W06, K_W07
W_03	Zna i rozumie podstawowe paradygmaty architektoniczne budowy systemu robota oraz potrafi podać ich przykłady.	K_W05. K_W06, K_W07
W_04	Wie z jakich komponentów zbudowany jest system nawigacyjny robota mobilnego,	K_W05. K_W06, K_W07
W_05	Zna i rozumie koncepcję agenta softwarowego oraz potrafi podać i scharakteryzować przykłady.	K_W05. K_W06, K_W07
Symbol efektu	Efekt uczenia się: UMIEJĘTNOŚCI	Symbol efektu kierunkowego

U_01	Potrafi pracować ze środowiskiem MS Robotics Studio,	K_U11
U_02	Potrafi budować prosty system nawigacyjny robota, bazujący na odruchach,	K_U15
U_03	Potrafi stosować algorytmy planowania tras,	K_U15, K_U17, K_U19
U_04	Potrafi zapewnić komunikację systemu robota z zewnętrznym systemem.	K_U15
Symbol efektu	<b>Efekt uczenia się: KOMPETENCJE SPOŁECZNE</b>	<b>Symbol efektu kierunkowego</b>
K_01	jest gotów do krytycznej oceny posiadanej wiedzy oraz jest gotów do uznawania znaczenia wiedzy w rozwiązywaniu problemów poznawczych i praktycznych z zakresu informatyki	K_K01
K_02	Potrafi formułować opinie na temat nowo-pojawiających się rozwiązań w obszarze robotyki. Rozumie rosnący udział systemów robotycznych w biznesie.	K_K02
<b>Forma i typy zajęć:</b>	Studia stacjonarne: wykłady (30 godz.), ćwiczenia laboratoryjne (45 godz.) Studia niestacjonarne: wykłady (18 godz.), ćwiczenia laboratoryjne (24 godz.)	
<b>Wymagania wstępne i dodatkowe:</b>		
1. Umiejętność programowania w języku C#, C++ i Java.		
<b>Treści modułu kształcenia:</b>		
<ol style="list-style-type: none"> <li><b>Podstawowy robotyki.</b> podstawowe pojęcia z dziedziny robotyki, przegląd zastosowań robotów, firmy zajmujące się produkcją i dystrybucją robotów.</li> <li><b>Przegląd środowisk programowania robotów mobilnych.</b> Player Stage, Orocos, MS Robotics Studio, URBI, Orocos, MROC++.</li> <li><b>Podstawowe paradygmaty architektoniczne układów sterujących robotami mobilnymi.</b> Paradygmat deliberatywny, reaktywny, behawioralny i hybrydowy. Klasyfikacje i przykłady systemów.</li> <li><b>Metodologie reaktywne.</b> Subsumcja, pola potencjałowe, mikro-zadania w ARIA.</li> <li><b>Podejścia hybrydowe.</b> AuRA, SFX, PRS, TCA, Saphira, CLARATy, Architektury trójwarstwowe,</li> <li><b>Podejścia agentowe.</b> Agent BDI, agent upostaciowiony, M-agent.</li> <li><b>Rozpoznawanie otoczenia za pomocą sensorów.</b> Sensory robota. Rozpoznawania cech środowiska na przykładzie wykrywania linii w skanie laserowym za pomocą algorytmu <i>RANSAC</i> lub <i>regresji liniowej</i>.</li> <li><b>Reprezentacje środowiska.</b> Typy i budowa map. Mapy metryczne – geometryczne i rastrowe. Mapy topologiczne. Mapy mieszane – topologiczno-metryczne.</li> <li><b>Problem samolokalizowania się robota.</b> Odometria, SLAM, Filtr Bayes'a, filtr Kalmana</li> <li><b>Algorytmy planowania tras.</b> Transformata dystansów, pola potencjałowe, metoda gradientowa, metoda grafu widoczności.</li> <li><b>Realizacja tras</b> - pilnowanie trajektorii, odruchy robota z wykorzystaniem funkcji reaktywnych.</li> <li><b>Systemy wielorobotowe.</b> Realizacja komunikacji i współpracy między robotami,</li> <li><b>Internet of Things</b> –DPWS (Devices Profile for Web Services), WS4D (Web Services for Devices), SODA (Service Oriented Device Architecture).</li> </ol>		

**Literatura podstawowa:**

1. Piotr Ciesielski, Jacek Sawoniewicz, Adam Szmigielski. Elementy robotyki mobilnej,
2. A. Ambroszkiewicz, G. Terlikowski. Inteligencja wokół nas: „Współdziałanie w systemach agentów softwareowych, mobilnych robotów oraz inteligentnych urządzeń”. Akademicka Oficyna Wydawnicza EXIT, 2010.
3. R. R. Murphy. „Introduction to AI Robotics”. MIT Press, Cambridge, MA, USA, 2000.

**Literatura dodatkowa:**

1. Zielińska Teresa. Maszyny kroczące. Wydawnictwo Naukowe PWN, 2012.
2. Małgorzata Kaliczyńska. Polskie innowacje w automatyce i robotyce, Warszawa 2013, Wydawca: Przemysłowy Instytut Automatyki i Pomiarów PIAP

**Planowane formy/działania/metody dydaktyczne:**

Wykład tradycyjny wspomagany technikami multimedialnymi, ćwiczenia praktyczne. Zamieszczanie na stronach internetowych materiałów do zajęć laboratoryjnych.

**Sposoby weryfikacji efektów kształcenia osiągniętych przez studenta:**

Efekty W\_01 – W\_05, K\_02 będą sprawdzane podczas pisemnego zaliczenia wykładu na ocenę. Przykładowe pytania o charakterze otwartym:

- Wymień i scharakteryzuj znane Ci paradygmaty układów sterujących robotami mobilnymi.
- Wymień i scharakteryzuj znane Ci podejścia do planowania tras.
- Wymień i scharakteryzuj znane Ci koncepcje do tworzenia systemów agentowych,
- Wymień i scharakteryzuj znane Ci protokoły umożliwiające realizację idei Internet of Things.

Efekt U\_01 - U\_04 oraz K1 - K2 będą systematycznie sprawdzane na zajęciach laboratoryjnych. Zadania na następne laboratorium muszą być dostępne co najmniej tydzień przed zajęciami. Student, na podstawie podanej literatury, musi się do nich samodzielnie lub korzystając z konsultacji przygotować.

Przykładowe zadania:

- Wykorzystując próbkę danych z dalmierza laserowego, wykryj w niej linie oraz ich przecięcia. Użyj algorytmu regresji liniowej.
- Posiadając przykładową mapę rastrową wyznacz w niej trasę z zadanego punktu początkowego do zadanego punktu docelowego wykorzystując algorytm transformaty dystansów.

**Forma i warunki zaliczenia:**

Moduł kończy się egzaminem. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie laboratorium. Na zaliczenie laboratorium składają się oceny cząstkowe uzyskane na regularnych zajęciach z nauczycielem akademickim oraz z samodzielnie wykonanego zadania indywidualnego według schematu:

- Regularne zajęcia – 26 pkt.,
- Obrona zadania indywidualnego – 14 pkt.

Zajęcia laboratoryjne będą zaliczone w wypadku uzyskania co najmniej połowy punktów z poszczególnych form aktywności studenta: regularne zajęcia – co najmniej 13 pkt., obrona indywidualnego zadania – co najmniej 7 pkt. Na tej formie zajęć student może maksymalnie uzyskać 40 pkt.

Egzamin jest egzaminem pisemno - ustnym. Można na nim uzyskać do 60 pkt. Egzamin będzie zaliczony w przypadku uzyskania co najmniej 30 pkt. Ocena końcowa z modułu (wystawiana po zaliczeniu

wszystkich części składowych), w zależności od sumy uzyskanych punktów (maksymalnie 100pkt.) jest następująca (w nawiasach ocena wg skali ECTS):

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

Poprawy:

Jednorazowa poprawa każdego ćwiczenia w trakcie zajęć w semestrze. Poprawa zadania indywidualnego w sesji egzaminacyjnej przed drugim terminem egzaminu.

Bilans punktów ECTS:

Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	30 godz.
Udział w ćwiczeniach laboratoryjnych	45 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	10 godz.
Udział w konsultacjach godz. z przedmiotu	5 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	10 godz.
Sumaryczne obciążenie pracą studenta	100 godz.
Punkty ECTS za przedmiot	4 ECTS

Studia niestacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	18 godz.
Udział w ćwiczeniach laboratoryjnych	24 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	36 godz.

Udział w konsultacjach godz. z przedmiotu	2 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	20 godz.
Sumaryczne obciążenie pracą studenta	100 godz.
Punkty ECTS za przedmiot	4 ECTS

<b>Sylabus przedmiotu / modułu kształcenia</b>		
<b>Nazwa przedmiotu/modułu kształcenia:</b>		<b>Bezpieczeństwo mobilnych systemów komputerowych</b>
<b>Nazwa w języku angielskim:</b>		Security of mobile computer systems
<b>Język wykładowy:</b>	polski	
<b>Kierunek studiów, dla którego przedmiot jest oferowany:</b>		informatyka
<b>Jednostka realizująca:</b>		Wydział Nauk Ścisłych i Przyrodniczych
<b>Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):</b>		fakultatywny
<b>Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):</b>		pierwszego stopnia
<b>Rok studiów:</b>	trzeci	
<b>Semestr:</b>	szósty	
<b>Liczba punktów ECTS:</b>	3	
<b>Imię i nazwisko koordynatora przedmiotu:</b>		dr Piotr Świtalski
<b>Imię i nazwisko prowadzących zajęcia:</b>		dr Piotr Świtalski
<b>Założenia i cele przedmiotu:</b>		Celem przedmiotu jest zaznajomienie studentów z problematyką bezpieczeństwa systemów komputerowych. Przedmiot ma za zadanie zapoznanie z klasycznymi technikami bezpieczeństwa w systemach komputerowych. W założeniach do tego przedmiotu przewiduje się zajęcia praktyczne z użyciem komputerów, podczas których studenci nabędą umiejętności z atakami na systemy komputerowe i również metody przeciwdziałania im. Dodatkowo studenci zapoznają się z problemami bezpieczeństwa w systemach mobilnych. Ma również usystematyzować wiedzę w zakresie standardów bezpieczeństwa.
<b>Symbol efektu</b>	<b>Efekt uczenia się: WIEDZA</b>	<b>Symbol efektu kierunkowego</b>
<b>W_01</b>	Zna i rozumie zagadnienia z zakresu bezpieczeństwa systemów komputerowych	<b>K_W05, K_W07</b>
<b>W_02</b>	Zna i rozumie standardy i uwarunkowania prawne związane z bezpieczeństwem systemów komputerowych	<b>K_W04, K_W10</b>



<b>W_03</b>	Zna i rozumie techniki związane z atakami na systemy komputerowe w tym z atakami na systemy mobilne i potrafi im przeciwdziałać	<b>K_W05, K_W07</b>
<b>W_04</b>	Zna i rozumie mechanizmy bezpieczeństwa komputerowego i potrafi je wykorzystać w przedstawionym środowisku	<b>K_W07, K_W12, K_W15</b>
<b>Symbol efektu</b>	<b>Efekt uczenia się: UMIEJĘTNOŚCI</b>	<b>Symbol efektu kierunkowego</b>
<b>U_01</b>	Potrafi sprawnie wyszukiwać w literaturze informacje związane z bezpieczeństwem systemów komputerowych, potrafi wyszukać informacje na temat nowych podatności wykrytych w systemach komputerowych	<b>K_U01</b>
<b>U_02</b>	Potrafi sprawnie wykorzystać narzędzia bezpieczeństwa systemów komputerowych	<b>K_U02, K_U17</b>
<b>U_03</b>	Potrafi dobrać adekwatne zabezpieczenia w stosunku do zagrożenia w systemie komputerowym	<b>K_U10, K_U17, K_U25</b>
<b>Forma i typy zajęć:</b>	studia stacjonarne: wykłady (30 godz.), ćwiczenia laboratoryjne (30 godz.) studia niestacjonarne: wykłady (15 godz.), ćwiczenia laboratoryjne (15 godz.)	
<b>Wymagania wstępne i dodatkowe:</b>		
Warunkiem uczestnictwa w zajęciach jest uprzednie zaliczenie następujących przedmiotów: „Architektura systemów komputerowych”, „Systemy operacyjne”, „Technologie sieciowe” lub znajomość literatury obowiązującej w tych przedmiotach.		
<b>Treści modułu kształcenia:</b>		
<ol style="list-style-type: none"> <li><b>Koncepcja bezpieczeństwa komputerowego.</b> Podstawowe zasady. Właściwości i klasyfikacja koncepcji bezpieczeństwa: integralność, dostępność, poufność, niezaprzeczalność, odpowiedzialność. Podstawowe pojęcia. Model bezpieczeństwa sieciowego. Minimalne standardy bezpieczeństwa. Polityka bezpieczeństwa.</li> <li><b>Przestępczość komputerowa.</b> Przesłpstwa ujawnione i nieujawnione. Obiekty, typy i sprawcy przestępstw. Piractwo komputerowe, sabotaż, wywiad gospodarczy, szpiegostwo, przestępstwa bankowe. Inżynieria społeczna. Phising. Inżynieria odwrotna. Organizacje przeciwdziałające przestępczości komputerowej.</li> <li><b>Aspekty prawne ochrony informacji i normalizacja.</b> Stan prawny ochrony informacji w Polsce. Ochrona danych osobowych. Hacking. Prawo autorskie a programy komputerowe. Normalizacja w zakresie bezpieczeństwa systemów informatycznych.</li> <li><b>Charakterystyka zagrożeń w systemach informatycznych.</b> Specyfika systemów informatycznych. Klasyfikacja zagrożeń. Zagrożenia związane z DNS. Złośliwe oprogramowanie. Sieci botnet. Statystyka typowych zagrożeń.</li> <li><b>Klasyczne techniki szyfrowania.</b> Dziedzina kryptografii i podstawowe pojęcia. Podstawowe techniki szyfrowania: technika podstawieniowa, szyfr Cezara, szyfry mono i polialfabetyczne,</li> </ol>		

- szyfr Playfaira, szyfr Vigenère'a. Techniki przestawieniowe, szyfr zygzakowy, maszyny wirnikowe. Szyfrowanie symetryczne. Ataki siłowe przeprowadzane na algorytmy szyfrowania.
6. **Szyfry strumieniowe i blokowe.** Szyfry strumieniowe. Szyfr strumieniowy RC4. Szyfry blokowe. Struktura i szyfr Feistela. Standard DES, efekt lawiny. Algorytm AES. Tryby operacyjne szyfrów blokowych. Szyfrowanie asymetryczne. Algorytm RSA.
  7. **Podpis cyfrowy.** Idea podpisu cyfrowego. Wymagania stawiane podpisom cyfrowym. Mechanizm uwierzytelniania komunikatów. Kody uwierzytelnienia komunikatów MAC. Podpis cyfrowy ElGamal. Standard DSS. Algorytm DSA. Kryptograficzne funkcje haszujące. Algorytm SHA-512. Paradoks urodzin.
  8. **Uwierzytelnianie.** Pojęcie uwierzytelniania. Uwierzytelnianie przez hasło. Strategie wyboru haseł. Inne metody uwierzytelniania. Protokoły uwierzytelniania: protokół challenge and response. Atak „człowiek pośrodku”. Dowód z wiedzą zerową. Uwierzytelnianie dwuskładnikowe. Hasło jednorazowe. Generowanie haseł jednorazowych – protokół S/KEY.
  9. **Szkodliwe oprogramowanie.** Problemy związane z nieautoryzowanym dostępem do systemów komputerowych. Infekcje systemów komputerowych. Bezpieczny system operacyjny. Podatność systemów operacyjnych. Charakterystyka szkodliwego oprogramowania: tylne drzwi, bomby logiczne, konie trojańskie, wirusy, robaki, exploity i rootkity, keyloggery. Przeciwdziałanie szkodliwemu oprogramowaniu.
  10. **Nadużycia w sieciach rozległych.** Protokoły komunikacyjne. Struktura nagłówek protokołów TCP/IP. Ataki na warstwę sieciową. Falszowanie pakietów IP. Fragmentacja pakietów IP. Ataki na warstwę transportową. Atak SYN Flood. Skanowanie portów. Ataki w sieciach komórkowych.
  11. **Zapory sieciowe (firewalle).** Model ogólny zapory sieciowej. Charakterystyka firewalli. Ograniczenia firewalli. Firewall filtrujący pakiety. Firewall filtrujący pakiety z badaniem stanu pakietu. Brama aplikacyjna, brama transmisyjna. Implementacja firewalla. Strefa zdemilitaryzowana (DMZ). Przykładowa konfiguracja firewalla z DMZ.
  12. **Systemy wykrywania intruzów.** Zachowania intruzywne. Wzorce zachowań intruzów. Wykrywanie intruzów. Statystyczna analiza zachowania. Wykrywanie intruzów w oparciu o reguły. Systemy IDPS. Audyt w systemach IDPS. Pułapki (Honeypoty).
  13. **Bezpieczeństwo systemów mobilnych.** Bezpieczeństwo infrastruktury mobilnej. Bezpieczeństwo urządzeń końcowych. Bezpieczeństwo aplikacji mobilnych. Zarządzanie bezpieczeństwem systemów mobilnych. Testy penetracyjne systemów mobilnych.
  14. **Bezpieczeństwo aplikacji mobilnych cz. 1.** Ataki w warstwie aplikacji. Przepelnienie bufora. Wstrzykiwanie kodu. Ataki SQL Injection. Błędy w uwierzytelnianiu użytkownika i zarządzania jego sesją. Atak XSS. Błędy w konfiguracji oprogramowania. Niewłaściwe zabezpieczenie wrażliwych danych. Atak Cross-Site Request Forgery (CSRF).
  15. **Bezpieczeństwo aplikacji mobilnych cz. 2.** Bezpieczeństwo systemu Android. Malware na urządzeniach mobilnych. Podatności przeglądarek internetowych. Modyfikacja zachowania aplikacji mobilnych. Inżynieria odwrotna aplikacji mobilnych. Hakowanie systemu Android. Zabezpieczanie systemu Android.

#### Literatura podstawowa:

1. Stallings W.: Kryptografia i bezpieczeństwo sieci komputerowych. Matematyka szyfrów i techniki kryptologii, Wyd. Helion, Gliwice, 2012.
2. Prasad P.: Testy penetracyjne nowoczesnych serwisów. Kompendium inżynierów bezpieczeństwa, Wyd. Helion, Gliwice, 2017.

3. Stuttard D., Pinto M., The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws, 2nd Edition, Wiley 2011.
4. Misra A., Dubey A., Android Security: Attacks and Defenses, Auerbach Publications, 2013.

#### Literatura dodatkowa:

1. Stallings W.: Kryptografia i bezpieczeństwo sieci komputerowych. Koncepcje i metody bezpiecznej komunikacji, Wyd. Helion, Gliwice, 2012.
2. Schetina E., Green K., Carlson J.: Bezpieczeństwo w sieci, Wyd. Helion, Gliwice, 2002.
3. Suehring S.: Zapory sieciowe w systemie Linux. Kompendium wiedzy o nftables. Wydanie IV, Wyd. Helion, Gliwice, 2015.

#### Planowane formy/działania/metody dydaktyczne:

Wykład tradycyjny wspomagany jest technikami multimedialnymi. Ćwiczenia laboratoryjne – zajęcia praktyczne z wykorzystaniem wybranych narzędzi programowych. Na stronie internetowej prowadzącego zamieszczane są materiały z problemami i zadaniami laboratoryjnymi.

#### Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:

Efekty W\_01 do W\_04 weryfikowane będą poprzez egzamin pisemny, a także w toku weryfikacji przygotowania do kolejnych zajęć laboratoryjnych. Na egzaminie pisemnym pytania będą dotyczyły poznanych technik ataków i sposobów zabezpieczania systemów komputerowych. Egzamin będzie również obejmował treści związane z kryptografią. Przykładowe pytania:

- Przedstaw schemat działania podpisu cyfrowego.
- Na czym polega uwierzytelnianie dwuskładnikowe?
- W jaki sposób przeprowadzany jest atak DDoS?
- Na czym polega szyfrowanie symetryczne?
- Omów mechanizmy bezpieczeństwa systemu Android.

Przed egzaminem studenci będą mieli dostęp do przykładowych pytań na egzamin.

Efekty U\_01 do U\_03 będą sprawdzane systematycznie na zajęciach laboratoryjnych. Przykładowe zadania:

- Zaimplementuj certyfikat SSL i dodaj go w wybranej przeglądarce internetowej.
- Napisz polecenie dla zapory iptables, które będzie odrzucało wszystkie nowe połączenia TCP z adresu IP o adresie 91.205.12.6 na porcie 22.

Materiały na następne laboratorium będą dostępne na dwa dni przed zajęciami.

#### Forma i warunki zaliczenia:

Ocena z przedmiotu składa się z dwóch ocen cząstkowych:

- oceny z zajęć laboratoryjnych,
- oceny z egzaminu końcowego.

Na ocenę z zajęć laboratoryjnych składają się oceny cząstkowe uzyskane na regularnych zajęciach z nauczycielem akademickim, za które można uzyskać sumarycznie 40 pkt. Zaliczenie zajęć laboratoryjnych możliwe po uzyskaniu co najmniej 51% liczby punktów z tej formy zaliczenia

Egzamin jest egzaminem pisemnym. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie laboratorium. Można na nim uzyskać maksymalnie 60 pkt. Egzamin będzie zaliczony w przypadku uzyskania co najmniej 51% liczby punktów z tej formy zaliczenia. Ocena końcowa z przedmiotu, w zależności od sumy uzyskanych punktów (maksymalnie 100 pkt) jest następująca (w nawiasach ocena wg skali ECTS):

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

#### **Bilans punktów ECTS:**

##### Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	30 godz.
Udział w ćwiczeniach laboratoryjnych	30 godz.
Przygotowanie się do egzaminu	8 godz.
Przygotowanie się do ćwiczeń laboratoryjnych	5 godz.
Udział w konsultacjach z przedmiotu	2 godz.
<b>Sumaryczne obciążenie pracą studenta</b>	<b>75 godz.</b>
<b>Punkty ECTS za przedmiot</b>	<b>3 ECTS</b>

##### Studia niestacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	15 godz.
Udział w ćwiczeniach laboratoryjnych	15 godz.
Przygotowanie się do egzaminu	20 godz.
Przygotowanie się do ćwiczeń laboratoryjnych	20 godz.
Udział w konsultacjach z przedmiotu	5 godz.

<b>Sumaryczne obciążenie pracą studenta</b>	<b>75 godz.</b>
<b>Punkty ECTS za przedmiot</b>	<b>3 ECTS</b>