

Politechnika Białostocka
Wydział Informatyki

Program studiów podyplomowych

**Informatyka - technologie ICT
i programowanie**

Białystok, kwiecień 2019 r.

PROGRAM STUDIÓW PODYPLOMOWYCH
INFORMATYKA – TECHNOLOGIE ICT I PROGRAMOWANIE

Studia podyplomowe **Informatyka – technologie ICT i programowanie** trwają **2** semestry i umożliwiają uzyskanie kwalifikacji cząstkowych na poziomie **6 PRK**. Łączna liczba punktów ECTS: **60**. Łączna liczba godzin zajęć: **300**.

Plan studiów
INFORMATYKA – TECHNOLOGIE ICT I PROGRAMOWANIE

Lp.	Nazwa przedmiotu	Kod	Liczba ECTS			Liczba godzin w semestrze			
			C	K	P	W	Ć	Ps	
SEMESTR I									
1	Wprowadzenie do architektury komputerów i systemów informatycznych	SPI001	6	1,5	4	16		16	zaliczenie z oceną
2	Bazy danych	SPI002	5	1	4	10		16	zaliczenie z oceną
3	Systemy operacyjne	SPI003	4	1	3,5	10		10	zaliczenie z oceną
4	Programowanie z elementami algorytmiki	SPI004	8	1,5	6	16		16	zaliczenie z oceną
5	Technologie sieciowe i standardy WWW	SPI005	6	1,5	4,5	16		16	zaliczenie z oceną
6	Techniki zapewniania jakości oprogramowania	SPI012	1	0,5	0	8		0	zaliczenie z oceną
RAZEM W SEMESTRZE			30	7	22	76		74	
SEMESTR II									
7	Programowanie obiektowe i zdarzeniowe	SPI006	7	1,5	5,5	16		16	zaliczenie z oceną
8	Inżynieria oprogramowania i analiza biznesowa	SPI007	6	1,5	4,5	16		16	zaliczenie z oceną
9	Programowanie aplikacji internetowych	SPI008	6	1,5	5,5	16		16	zaliczenie z oceną
10	Bezpieczeństwo systemów informatycznych	SPI009	4	1	3,5	10		10	zaliczenie z oceną
11	Wprowadzenie do biometrii	SPI014	2	1	1,5	8		10	zaliczenie z oceną
12	Projektowanie UX i interfejsów użytkownika	SPI010	5	0,5	4,5	8		8	zaliczenie z oceną
RAZEM W SEMESTRZE			30	7	25	74		76	
	RAZEM KIERUNEK [w podziale na FORMY ZAJĘĆ]								
ŁĄCZNIE ECTS/GODZ. W TRAKCIE STUDIÓW			60	14	47	150		150	

Objaśnienia do punktów ECTS: C – Całkowita wartość punktowa, K – Punkty kontaktowe, P – Punkty praktyczne

Sylwetka absolwenta

Studia podyplomowe na kierunku „Informatyka – technologie ICT i programowanie” przeznaczone są dla wszystkich absolwentów szkół wyższych, którzy pragną zdobyć dodatkowe kwalifikacje, zaktualizować posiadaną już wiedzę, czy zmienić dotychczas wykonywany zawód. Głównym celem kształcenia na kierunku Informatyka jest uzyskanie wiedzy informatycznej, pozwalającej na wejście w świat Informatyki i poznanie podstaw tworzenia rozwiązań informatycznych w zakresie aplikacji komputerowych i serwisów internetowych.

Celem studiów jest wprowadzanie słuchaczy w świat zagadnień systemów komputerowych i dedykowanego oprogramowania narzędziowego i systemowego. Program studiów jest przygotowany głównie z myślą o zdobyciu wiedzy związanej z programowaniem, działaniem systemów operacyjnych, narzędzi programistycznych i baz danych. Najważniejszym elementem studiów jest nauka programowania obiektowego i zagadnień z tym związanych, od podstaw. Dopelnieniem materiału jest wiedza z zakresu sieci komputerowych, technologii sieciowych i standardów WWW.

Studia podyplomowe na kierunku „Informatyka - technologie ICT i programowanie” przeznaczone są dla wszystkich absolwentów szkół wyższych, którzy pragną zdobyć dodatkowe kwalifikacje, zaktualizować posiadaną już wiedzę, czy zmienić dotychczas wykonywany zawód. Głównym celem kształcenia na kierunku Informatyka jest uzyskanie wiedzy informatycznej, pozwalającej na wejście w świat informatyki i wprowadzenie do programowania.

Absolwenci tych studiów będą mogli podjąć pracę w firmach, organizacjach czy jednostkach administracji wykorzystujących systemy informatyczne. Wiedza zdobyta w czasie studiów to także pierwszy krok, który może pozwolić na pracę w firmach wytwarzających i wdrażających rozwiązania informatyczne.

Opis kompetencji oczekiwanych od kandydata ubiegającego się o przyjęcie na studia podyplomowe

Uczestnikiem studiów podyplomowych może być osoba, która posiada kwalifikację pełną co najmniej na poziomie 6 uzyskaną w systemie szkolnictwa wyższego i nauki.

Kandydaci ubiegający się o przyjęcie na studia podyplomowe powinni mieć podstawową wiedzę i umiejętności z zakresu obsługi komputera i urządzeń peryferyjnych oraz znajomość podstawowych zagadnień związanych z technologiami informacyjnymi.

Zestawienie efektów uczenia się

Zestawienie tabelaryczne kierunkowych efektów uczenia się odnoszących się do charakterystyk drugiego stopnia określonych na podstawie ustawy z dnia 22 grudnia 2015 r. o zintegrowanym systemie kwalifikacji na poziomie 6 PRK

Objaśnienia oznaczeń:

P6 – poziom 6 PRK (Polskie Ramy Kwalifikacji)

S – charakterystyka typowa dla kwalifikacji uzyskiwanych w ramach szkolnictwa wyższego

W – wiedza

T – teorie, zasady

Z – zjawiska i procesy
O – organizacja pracy
G – głębia i zakres
K – kontekst
U – umiejętności
I – informacje
W – wykorzystanie wiedzy
K – komunikowanie się
O – organizacja pracy
U – uczenie się
K – kompetencje społeczne
K – krytyczna ocena
O – odpowiedzialność
R – rola zawodowa
INF – Informatyka – technologie ICT i programowanie
1, 2, 3 i kolejne – numery efektu uczenia się

Załącznik do Uchwały Senatu PB nr 328/XVIII/XV/2018
Załącznik nr 1 do „Wytyczne do tworzenia programów studiów podyplomowych”

Symbol	Efekty uczenia się dla studiów podyplomowych	Odniesienie do charakterystyk drugiego stopnia określonych na podstawie art. 7 ust. 3 Ustawy z dnia 22 grudnia 2015 r. o Zintegrowanym Systemie Kwalifikacji na poziomie 6 PRK	Odniesienie do charakterystyk drugiego stopnia określonych na podstawie art. 7 ust. 4 Ustawy z dnia 22 grudnia 2015 r. o Zintegrowanym Systemie Kwalifikacji na poziomie 6 PRK
Wiedza: absolwent zna i rozumie			
INF_W01	pojęcia i standardy związane z architekturą komputerów, systemami operacyjnymi, sieciami komputerowymi i bezpieczeństwem systemów informatycznych	P6S_WG	P6Z_WT
INF_W02	konstrukcje, funkcje i zastosowanie wybranego języka programowania, wybranych algorytmów i programowania obiektowego	P6S_WG	P6Z_WO
INF_W03	pojęcia dotyczące projektowania interfejsów użytkownika oraz wytycznych w User Experience	P6S_WG, P6S_WK	P6Z_WT
INF_W04	zagadnienia analizy wymagań, zarządzania projektem informatycznym, inżynierii oprogramowania	P6S_WG	P6Z_WT
INF_W05	zasady budowy serwisów WWW, aplikacji internetowych, baz danych oraz technologii i standardów sieciowych i WWW	P6S_WG	P6Z_WT, P6Z_WZ
Umiejętności: absolwent potrafi			
INF_U01	posługiwać się podstawowymi poleceniami i narzędziami systemu operacyjnego MS Windows i Linux oraz konfigurować wybrane usługi sieci komputerowych samodzielnie i w zespole.	P6S_UW, P6S_UO	P6Z_UO
INF_U02	programować na poziomie podstawowym wykorzystując proste konstrukcje oraz funkcje języka obiektowego w wybranych zagadnieniach.	P6S_UW	P6Z_UI, P6Z_UN
INF_U03	analizować problem i zastosować wytyczne dotyczące projektowania interfejsów i User Experience przy budowie aplikacji oraz rozwijać te umiejętności	P6S_UW, P6S_UU	P6Z_UN
INF_U04	wykonać i przedyskutować analizę i projekt prostego systemu komputerowego, pracować zespołowo i wykorzystywać w pracy wybrane narzędzia.	P6S_UW, P6S_UO, P6S_UK	P6Z_UI
INF_U05	wykorzystać zasady budowy aplikacji internetowych oraz standardów WWW do budowy responsywnych serwisów internetowych oraz rozwijać te umiejętności	P6S_UW, P6S_UU	P6Z_UN
INF_U06	wykorzystać znane metody biometryczne w realnych systemach oraz przedstawić pisemnie lub ustnie poprawność ich budowy	P6S_UW, P6S_UK	P6Z_UI
Kompetencje społeczne: absolwent jest gotów do			
INF_S01	krytycznej oceny posiadanej wiedzy	P6S_KK	P6Z_KP
INF_S02	wejścia na rynek pracy ICT i inicjowania działań zwiększających wykorzystanie ICT	P6S_KO	P6Z_KP, P6Z_KW
INF_S03	przestrzegania zasad etyki w zawodzie informatyka i dbania o dorobek informatyki	P6S_KR	P6Z_KW, P6Z_KO

Matryca efektów uczenia się

Załącznik do Uchwały Senatu PB nr 328/XVIII/XV/2018

Załącznik nr 2 do Wyttyczne do tworzenia programów studiów podyplomowych

				Nazwa studiów podyplomowych: Informatyka - technologie ICT i programowanie															
				MATRYCA POKRYCIA EFEKTÓW UCZENIA SIĘ															
				WIEDZA						UMIEJĘTNOŚCI						KOM. SPOŁ.			
Lp.	Nazwa przedmiotu	Kod przedmiotu	semestr	INF_W01	INF_W02	INF_W03	INF_W04	INF_W05	Kod przedmiotu	INF_U01	INF_U02	INF_U03	INF_U04	INF_U05	INF_U06	INF_S01	INF_S02	INF_S03	Kod przedmiotu
1	Wprowadzenie do architektury komputerów i systemów informatycznych	SPI001	pierwszy	+					SPI001		+								SPI001
2	Bazy danych	SPI002						+	SPI002					+					SPI002
3	Systemy operacyjne	SPI003		+					SPI003	+									SPI003
4	Programowanie z elementami algorytmiki	SPI004			+				SPI004		+								SPI004
5	Technologie sieciowe i standardy WWW	SPI005		+					SPI005	+							+		SPI005
6	Współczesne technologie ICT i inżynieria danych	SPI012				+			SPI012			+							SPI012
7	Programowanie obiektowe i zdarzeniowe	SPI006	drugi		+				SPI006		+						+		SPI006
8	Inżynieria oprogramowania i analiza biznesowa	SPI007					+		SPI007				+					+	SPI007
9	Programowanie aplikacji internetowych	SPI008						+	SPI008					+					SPI008
10	Bezpieczeństwo systemów informatycznych	SPI009		+					SPI009	+									SPI009
11	Projektowanie UX i interfejsów użytkownika	SPI010				+			SPI010			+							SPI010
12	Wprowadzenie do biometrii	SPI014		+					SPI014		+					+			
suma:				5	2	2	1	2		3	4	2	1	2	1	1	1	1	

Zasoby biblioteczne

- A. Koszljada, Zarządzanie projektami IT. Przewodnik po metodykach, Helion, 2010.
- A. Roman, K. Zmitrowicz, Testowanie oprogramowania w praktyce. Studium przypadków, Wydawnictwo Naukowe PWN, 2016.
- A. Roman, Testowanie i jakość oprogramowania. Modele, techniki, narzędzia, Wydawnictwo Naukowe PWN, 2017.
- A. S. Tanenbaum, Modern Operating Systems, Pearson Education, 2015.
- A. S. Tanenbaum, Sieci komputerowe, Helion, 2004.
- A. Silberschatz, P.B. Galvin, G. Gagne, Podstawy systemów operacyjnych, Wydawnictwa Naukowo-Techniczne, 2006.
- A. Skorpiski, Podstawy budowy i działania komputerów, WKiŁ, 2000.
- A. Stellman, C#, Helion, 2011.
- B. Eisenberg, J. Quarto-von Tivadar, B. Crosby, L. T. David, Google Website Optimizer. Przewodnik, Helion, 2010
- B. Keringhan, D. Ritchie, Język ANSI C. Programowanie, Helion, 2010.
- B. Pfaffenberger, S. M. Schafer, C. White, B. Karow, HTML, XHTML i CSS. Biblia, Helion, 2005.
- D. E. Comer, Sieci komputerowe i intersieci, Wydawnictwa Naukowo-Techniczne, 2003.
- D. Esposito, Microsoft ASP.NET and AJAX: architecting Web applications, Microsoft Press, 2009.
- D. Harel, Y. Feldman, Rzecz o istocie informatyki. Algorytmika, WNT, 2009.
- D. Pilone, N. Pitman, UML 2.0. Almanach, Helion, O'Reilly, 2007.
- J. Beaird, Niezawodne zasady web designu. Projektowanie spektakularnych witryn internetowych, Helion, 2012
- J. Biernat, Arytmetyka komputerów, Wydawnictwa Naukowe PWN, 1996.
- J. D. Ullman, J. Widom, Podstawowy wykład z systemów baz danych, WNT, 2001.
- J. Pieprzyk, T. Hardjono, J. Seberry, Teoria bezpieczeństwa systemów komputerowych, Helion, 2005.
- K. Sacha, Inżynieria oprogramowania, PWN, 2010.
- K. Subieta, Wprowadzenie do inżynierii oprogramowania, Wydawnictwo PJWSTK, 2002.
- K. Ślot, Rozpoznawanie biometryczne: nowe metody ilościowej reprezentacji obiektów, Wydawnictwa Komunikacji i Łączności, 2010.
- K. Ślot, Wybrane zagadnienia biometrii, Wydawnictwa Komunikacji i Łączności, 2008.
- L. Banachowski, A. Chądzyńska, E. Mrówka-Matejewska, K. Matejewski, K. Stencel, Bazy danych. Wykłady i ćwiczenia, Wydawnictwo PJWSTK, 2003.
- M. Fairhurst, Biometrics: A Very Short Introduction, Oxford Press, 2019.
- M. MacDonald, Pro WPF in C# 2010: Windows Presentation Foundation in .NET 4, Apress, 2010.
- P. Beynon-Davies, Systemy baz danych, WNT, 1998.
- P. Ciccarelli, C. Faulkner, Sieci: podstawy, Mikom, 2005.
- P. Kim, Podręcznik pentestera: bezpieczeństwo systemów informatycznych, Helion, 2015.
- P. Metzger, Anatomia PC: Kompendium, Helion, 2008.
- R. Bolle et. al., Biometria, Wydawnictwa Naukowo-Techniczne, 2008.
- R. C. Martin, Czysty kod. Podręcznik dobrego programisty, Helion, 2014.
- R. Elmasri, S. B. Navathe, Wprowadzenie do systemów baz danych, Helion, 2005.

- R. K. Wysocki, Efektywne zarządzanie projektami, Helion, 2013.
- S. Krug, Nie każ mi myśleć! O życiowym podejściu do funkcjonalności stron internetowych, Helion, 2006
- S. Prata, Język C. Szkoła programowania, Helion, 2006.
- T. Trejderowski, Kradzież tożsamości: terroryzm informatyczny: cyberprzestępstwa, internet, telefon, facebook, ENETEIA Wydaw. Psychologii i Kultury, 2013.
- W. Dudek, Bazy danych SQL. Teoria i praktyka, Helion, 2005.
- W. Stallings, Systemy operacyjne, Wydawnictwo „Robomatic”, 2004.

Wszystkie powyższe pozycje dostępne są w Bibliotece PB.

Elektroniczne zasoby wiedzy

- Dokumentacje techniczne dostępne w Internecie (dokumenty RFC, instrukcje do oprogramowania itp..)
- <http://uxmag.com>
- <http://uxbite.com>
- <http://online-behavior.com>
- <http://www.getelastic.com>
- <http://blog.kissmetrics.com>
- Dokumentacja systemu Linux
- Materiały szkoleniowe z kursu Microsoft IT Academy "Aplikacje Internetowe"

Ramowe programy przedmiotów

Karty przedmiotów zgodne ze wzorem - Załącznik nr 1 do Zarządzenia Nr 915 z 2019 r. Rektora PB

Wydział Informatyki										
Kierunek studiów	Informatyka - technologie ICT i programowanie							Poziom i forma studiów	studia podyplomowe	
Specjalność / ścieżka dyplomowania								Profil kształcenia		
Nazwa przedmiotu	Wprowadzenie do architektury komputerów i systemów informatycznych							Kod przedmiotu	SPI001	
								Rodzaj przedmiotu	obowiązkowy	
Formy zajęć i liczba godzin	W	C	L	P	Ps	T	S	Semestr	1	
	16				16			Punkty ECTS	6	
Przedmioty wprowadzające										
Cele przedmiotu	Słuchacz uzyska wiedzę na temat: sposobów reprezentacji liczb, budowy komputera i jego bloków funkcjonalnych, budowy i działania mikroprocesora czy systemów operacyjnych oraz podstawowych pojęć związanych z sieciami komputerowymi. Po zakończeniu kursu Słuchacz powinien umieć napisać prosty program w języku niskiego poziomu, skonfigurować system operacyjny oraz skonfigurować połączenia sieciowe w systemie operacyjnym.									
Treści programowe	Wykład: Podstawowe pojęcia. Systemy liczbowe. Zapis uzupełnień do 2 i podstawowe działania w tym zakresie. Reprezentacje zmiennoprzecinkowe liczb i podstawowe działania w tym zakresie. Koncepcja J. Von Neumanna maszyny z pamiętany programem. Podstawowe bloki komputera, ich parametry i technologie. Organizacja i funkcjonowanie mikrokomputera IBM PC. Podstawowe rejestry mikroprocesora Intel 8086, tryb adresów rzeczywistych, wybrane rozkazy. Mapa pamięci. Przerwania. Elementy organizacji systemu operacyjnego. Organizacja i implementacja systemu plików. Komunikacja z urządzeniami peryferyjnymi. Interfejsy komputera. Podstawowe pojęcia sieci komputerowych. Sieć lokalna, technologia Ethernet. Sieć rozległa Internet: adresy komputerów, nazwy domenowe, protokoły TCP/IP. Pracownia specjalistyczna: Systemy liczbowe. Zapis uzupełnień do 2 i podstawowe działania w tym zakresie. Reprezentacje zmiennoprzecinkowe liczb i podstawowe działania w tym zakresie. Język niskiego poziomu – assembler. Podstawowe rejestry mikroprocesora Intel 8086, tryb adresów rzeczywistych, wybrane rozkazy. Mapa pamięci. Przerwania. Tworzenie programu w assemblerze.									
Metody dydaktyczne	1. prezentacja multimedialna; 2. analiza wybranych przypadków; 3. samodzielne opracowanie wskazanych zagadnień; 4. ćwiczenia indywidualne, ćwiczenia grupowe; 5. przygotowanie przez słuchacza samodzielnej koncepcji projektowej; 6. dyskusje dydaktyczne.									
Forma zaliczenia	Wykład – kolokwium, Pracownia specjalistyczna – projekt końcowy lub projekty cząstkowe na zajęciach									
Symbol efektu uczenia się	Zakładane efekty uczenia się								Odniesienie do kierunkowych efektów uczenia się	
EU1	Zna sposoby reprezentacji liczb w systemach cyfrowych								INF_W01	
EU2	Zna budowę i zasadę działania komputera i jego bloków funkcjonalnych								INF_W01	
EU3	Zna budowę i zasadę działania systemu operacyjnego								INF_W01	
EU4	Potrafi wykorzystywać różne formy zapisu liczb w systemach								INF_U02	

	cyfrowych w programowaniu	
EU5	Potrafi obsługiwać system operacyjny w stopniu umożliwiającym jego nietrywialną konfigurację	INF_U02
Symbol efektu uczenia się	Sposoby weryfikacji efektów uczenia się	Forma zajęć, na której zachodzi weryfikacja
EU1	Kolokwium	W
EU2	Kolokwium	W
EU3	Kolokwium	W
EU4	Projekt końcowy lub projekty cząstkowe na zajęciach	Ps
EU5	Projekt końcowy lub projekty cząstkowe na zajęciach	Ps
Bilans nakładu pracy słuchacza (w godzinach)		Liczba godz.
Wyliczenie	Udział w wykładach	16
	Udział w pracowni specjalistycznej	16
	Przygotowanie do zajęć pracowni specjalistycznej	30
	Wykonanie zadań domowych (prac domowych)	30
	Udział w konsultacjach związanych z projektem	2
	Realizacja zadań projektowych	30
	Przygotowanie do zaliczenia wykładu i obecność na nim	30
RAZEM:		154
Wskaźniki ilościowe		GODZINY ECTS
Nakład pracy słuchacza związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela		34 1,5
Nakład pracy słuchacza związany z zajęciami o charakterze praktycznym		108 4
Literatura podstawowa	1. A. Skorupski, Podstawy budowy i działania komputerów, WKiŁ, 2000. 2. W. Stallings, Systemy operacyjne, Wydawnictwo „Robomatic”, 2004. 3. J. Biernat, Arytmetyka komputerów, Wydawnictwa Naukowe PWN, 1996. 4. P. Ciccarelli, C. Faulkner, Sieci: podstawy, „Mikom”, 2005. 5. P. Metzger, Anatomia PC: Kompendium, Helion, 2008.	
Literatura uzupełniająca	1. O. Kokoreva, Windows XP: Rejestr systemu, Helion, 2005. 2. J. Mrugalski, ABC ochrony komputera przed atakami hakera, Helion, 2003. 3. R. J. Hantanon, Bezpieczeństwo systemu Linux, „Mikom”, 2002. 4. A. Simpson, Windows XP PL: Biblia, Helion, Gliwice 2003.	
Jednostka realizująca	Wydział Informatyki Politechniki Białostockiej	Data opracowania programu
Program opracował	<i>dr inż. Adam Klimowicz, dr inż. Paweł Tadejko</i>	11.04.2019

Wydział Informatyki										
Kierunek studiów	Informatyka - technologie ICT i programowanie							Poziom i forma studiów	studia podyplomowe	
Specjalność / ścieżka dyplomowania								Profil kształcenia		
Nazwa przedmiotu	Bazy danych							Kod przedmiotu	SPI002	
								Rodzaj przedmiotu	obowiązkowy	
Formy zajęć i liczba godzin	W	C	L	P	Ps	T	S	Semestr	1	
	10				16			Punkty ECTS	5	
Przedmioty wprowadzające										
Cele przedmiotu	Zadaniem zajęć jest zapoznanie słuchaczy z relacyjnymi bazami danych, językiem SQL oraz działaniem systemów zarządzania bazami danych. Celem przedmiotu jest zdobycie przez słuchacza wiedzy i umiejętności z zakresu budowy schematu relacyjnej bazy danych oraz formułowania zapytań w języku SQL.									
Treści programowe	Wykład: Wprowadzenie do baz danych i relacyjnych baz danych; Normalizacja relacji; Zapytania: proste, grupujące, zagnieżdżone, podzapytania skorelowane, z połączeniem zewnętrznym; Diagramy związków encji ERD i przejście od diagramów ER do modelu relacyjnego; Fizyczna organizacja danych w bazie danych; Transakcje; Operatory UNION, INTERSECT, MINUS; Język DDL i DML; Optymalizacja zapytań. Pracownia specjalistyczna: Zapytania: proste, grupujące, zagnieżdżone, podzapytania skorelowane, z połączeniem zewnętrznym; Diagramy związków encji ERD i przejście od diagramów ER do modelu relacyjnego; Transakcje; Operatory UNION, INTERSECT, MINUS; Język DDL i DML; Optymalizacja zapytań.									
Metody dydaktyczne	1. prezentacja multimedialna; 2. analiza wybranych przypadków; 3. samodzielne opracowanie wskazanych zagadnień; 4. ćwiczenia indywidualne, ćwiczenia grupowe; 5. przygotowanie przez słuchacza samodzielnej koncepcji projektowej; 6. dyskusje dydaktyczne.									
Forma zaliczenia	Wykład – kolokwium, Pracownia specjalistyczna – projekt końcowy lub projekty częściowe na zajęciach, obrona projektów									
Symbol efektu uczenia się	Zakładane efekty uczenia się								Odniesienie do kierunkowych efektów uczenia się	
EU1	Ma uporządkowaną i podbudowaną teoretycznie wiedzę w zakresie baz danych								INF_W05	
EU2	Ma wiedzę w zakresie projektowania i konstruowania bazy danych								INF_W05	
EU3	Potrafi konstruować zapytania SQL do dowolnej relacyjnej bazy danych								INF_U05, INF_S01	
EU4	Potrafi utworzyć schemat relacyjnej bazy danych								INF_U05, INF_S01	
Symbol efektu uczenia się	Sposoby weryfikacji efektów uczenia się								Forma zajęć, na której zachodzi weryfikacja	
EU1	Kolokwium								W	
EU2	Kolokwium								W	
EU3	Projekt końcowy lub projekty częściowe na zajęciach, obrona projektów								Ps	

EU4	Projekt końcowy lub projekty cząstkowe na zajęciach, obrona projektów	Ps	
Bilans nakładu pracy słuchacza (w godzinach)		Liczba godz.	
Wyliczenie	Udział w wykładach	10	
	Udział pracowni specjalistycznej	16	
	Przygotowanie do pracowni specjalistycznej	30	
	Wykonanie zadań domowych (prac domowych)	40	
	Udział w konsultacjach związanych z pracownią specjalistyczną	2	
	Realizacja zadań projektowych (w tym przygotowanie prezentacji)	15	
	Przygotowanie do kolokwium i obecność na nim	15	
	Przygotowanie do zaliczenia pracowni specjalistycznej (obrony projektów) i obecność na zaliczeniach (obronach)	17	
	RAZEM:	145	
Wskaźniki ilościowe		GODZINY	ECTS
Nakład pracy słuchacza związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela		28	1
Nakład pracy słuchacza związany z zajęciami o charakterze praktycznym		120	4
Literatura podstawowa	1. J. D. Ullman, J. Widom, Podstawowy wykład z systemów baz danych, WNT, 2001. 2. W. Dudek, Bazy danych SQL. Teoria i praktyka, Helion, 2005. 3. L. Banachowski, A. Chądryńska, E. Mrówka-Matejewska, K. Matejewski, K. Stencel, Bazy danych. Wykłady i ćwiczenia, Wydawnictwo PJWSTK, 2003. 4. P. Beynon-Davies, Systemy baz danych, WNT, 1998. 5. R. Elmasri, S. B. Navathe, Wprowadzenie do systemów baz danych, Helion, 2005.		
Literatura uzupełniająca	1. L. Banachowski, Bazy danych. Tworzenie aplikacji, PLJ, 1998. 2. H. Garcia-Molina, J. D. Ullman, J. Widom, Systemy baz danych. Pełny wykład, WNT, 2006.		
Jednostka realizująca	Wydział Informatyki Politechniki Białostockiej	Data opracowania programu	
Program opracował(a)	<i>dr inż. Małgorzata Krętowska, dr inż. Marcin Czajkowski</i>	11.04.2019	

Wydział Informatyki										
Kierunek studiów	Informatyka - technologie ICT i programowanie							Poziom i forma studiów	studia podyplomowe	
Specjalność / ścieżka dyplomowania								Profil kształcenia		
Nazwa przedmiotu	Systemy operacyjne							Kod przedmiotu	SPI003	
								Rodzaj przedmiotu	obowiązkowy	
Formy zajęć i liczba godzin	W	C	L	P	Ps	T	S	Semestr	1	
	10				10			Punkty ECTS	4	
Przedmioty wprowadzające										
Cele przedmiotu	Słuchacz zdobędzie wiedzę z podstaw architektury systemów operacyjnych oraz bezpieczeństwa i administracji systemów operacyjnych na przykładzie systemu Linux/MS Windows. Ponadto słuchacz zdobędzie umiejętności wykorzystania interfejsu tekstowego w zadaniach administracyjnych w systemie Linux/MS Windows, implementacji prostych skryptów w języku powłoki Bash, administrowania kontami/grupami użytkowników w systemie Linux/MS Windows.									
Treści programowe	Wykład: Elementy architektury systemów operacyjnych. Zarządzanie systemami plików. Mechanizmy koordynacji współpracy procesów i wątków. Podstawowe polecenia i ochrona zasobów w systemie Linux. Administracja użytkownikiem w systemie Linux/Windows. Skrypty w języku powłoki Bash/ MS Windows PowerShell. Strumieniowe przetwarzanie tekstu. Pracownia specjalistyczna: System operacyjny. Wiersz poleceń. Podstawowe polecenia. System plików i katalogów. Ochrona zasobów. Zarządzanie użytkownikami. Przetwarzanie strumieniowe.									
Metody dydaktyczne	1. prezentacja multimedialna; 2. analiza wybranych przypadków; 3. samodzielne opracowanie wskazanych zagadnień; 4. ćwiczenia indywidualne, ćwiczenia grupowe; 5. przygotowanie przez słuchacza samodzielnej koncepcji projektowej; 6. dyskusje dydaktyczne.									
Forma zaliczenia	Wykład – kolokwium, Pracownia specjalistyczna – projekt końcowy lub projekty częściowe na zajęciach									
Symbol efektu uczenia się	Zakładane efekty uczenia się								Odniesienie do kierunkowych efektów uczenia się	
EU1	Zna podstawowe zadania systemu operacyjnego								INF_W01	
EU2	Zna podstawowe mechanizmy zabezpieczania danych w systemie Linux/MS Windows								INF_W01	
EU3	Potrafi wykorzystać podstawowe polecenia powłoki systemu Linux/MS Windows PowerShell								INF_U01	
EU4	Realizuje proste czynności administracji użytkownikiem w systemie Linux/MS Windows samodzielnie i zespołowo								INF_U01	
Symbol efektu uczenia się	Sposoby weryfikacji efektów uczenia się								Forma zajęć, na której zachodzi weryfikacja	
EU1	Kolokwium zaliczające wykład								W	
EU2	Kolokwium zaliczające wykład								W	
EU3	Wykonanie zadania na zajęciach								Ps	

EU4	Wykonanie zadania na zajęciach	Ps	
Bilans nakładu pracy słuchacza (w godzinach)		Liczba godz.	
Wyliczenie	Udział w wykładach	10	
	Udział w pracowni specjalistycznej	10	
	Przygotowanie do pracowni specjalistycznej	20	
	Opracowanie sprawozdań z pracowni specjalistycznej i wykonanie zadań domowych	48	
	Udział w konsultacjach związanych z projektami	2	
	Realizacja zadań projektowych	12	
	Udział w kolokwium	2	
	RAZEM:	104	
Wskaźniki ilościowe		GODZINY	ECTS
Nakład pracy słuchacza związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela		24	1
Nakład pracy słuchacza związany z zajęciami o charakterze praktycznym		92	3,5
Literatura podstawowa	1. A. Silberschatz, P.B. Galvin, G. Gagne, Podstawy systemów operacyjnych, Wydawnictwa Naukowo-Techniczne, 2006. 2. A. S. Tanenbaum, Modern Operating Systems, Pearson Education, 2015. 3. Dokumentacja systemu Linux.		
Literatura uzupełniająca	1. W. R. Stevens, Advanced Programming in the UNIX Environment, Addison Wesley, 2013. 2. H. Schwichtenberg, Windows PowerShell. Podstawy, Helion, 2009.		
Jednostka realizująca	Wydział Informatyki Politechniki Białostockiej	Data opracowania programu	
Program opracował	dr inż. Ireneusz Mrozek	11.04.2019	

Wydział Informatyki										
Kierunek studiów	Informatyka - technologie ICT i programowanie							Poziom i forma studiów	studia podyplomowe	
Specjalność / ścieżka dyplomowania								Profil kształcenia		
Nazwa przedmiotu	Programowanie z elementami algorytmiki							Kod przedmiotu	SPI004	
								Rodzaj przedmiotu	obowiązkowy	
Formy zajęć i liczba godzin	W	C	L	P	Ps	T	S	Semestr	1	
	16				16			Punkty ECTS	8	
Przedmioty wprowadzające										
Cele przedmiotu	Zapoznanie słuchaczy z podstawowymi technikami programowania strukturalnego. Słuchacze nauczą się praktycznego stosowania tych technik w prostych programach, projektowania i implementowania prostych programów z wykorzystaniem programowania strukturalnego, wykształcą podstawowe umiejętności oceny efektywności algorytmu oraz projektowania efektywnych rozwiązań jeszcze przed fazą implementacji algorytmu.									
Treści programowe	Wykład: Algorytm. Programowanie: algorytmizacja, kodowanie. Schematy blokowe. Programowanie strukturalne. Symbole podstawowe. Podstawowe struktury danych: stałe, standardowe typy skalarne. Operatory. Wyrażenia. Definiowanie zmiennych, podstawowe instrukcje. Instrukcje strukturalne. Iteracje. Definiowanie typów. Tablice. Instrukcja wyboru. Funkcje. Przekazywanie parametrów. Struktura programu, typy strukturalne: struktury, napisy. Pliki elementowe. Pliki tekstowe. Formatowanie wyjścia. Typ wskaźnikowy. Rekurencja jako technika kodowania algorytmów. Złożoność obliczeniowa algorytmu. Pracownia specjalistyczna: Programowanie, kodowanie. Programowanie strukturalne. Symbole podstawowe. Podstawowe struktury danych: stałe, standardowe typy skalarne. Operatory. Wyrażenia. Definiowanie zmiennych, podstawowe instrukcje. Instrukcje strukturalne. Iteracje. Definiowanie typów. Tablice. Instrukcja wyboru. Funkcje. Przekazywanie parametrów. Struktura programu, typy strukturalne: struktury, napisy. Pliki elementowe. Pliki tekstowe. Formatowanie wyjścia. Typ wskaźnikowy. Rekurencja jako technika kodowania algorytmów.									
Metody dydaktyczne	1. prezentacja multimedialna; 2. analiza wybranych przypadków; 3. samodzielne opracowanie wskazanych zagadnień; 4. ćwiczenia indywidualne, ćwiczenia grupowe; 5. przygotowanie przez słuchacza samodzielnej koncepcji projektowej; 6. dyskusje dydaktyczne.									
Forma zaliczenia	Wykład – kolokwium, Pracownia specjalistyczna – projekt końcowy lub projekty częściowe na zajęciach									
Symbol efektu uczenia się	Zakładane efekty uczenia się								Odniesienie do kierunkowych efektów uczenia się	
EU1	Posiada podstawową wiedzę w zakresie projektowania i programowania strukturalnego prostych aplikacji								INF_W02	
EU2	Zna fundamentalne pojęcia i notacje stosowane w analizie algorytmów i struktur danych oraz w opisywaniu problemów obliczeniowych, decyzyjnych itp., które można rozwiązać za pomocą komputera								INF_W02	
EU3	Potrafi przetestować program strukturalny. W przypadku wykrycia								INF_U02	

	błędów potrafi przeprowadzić ich diagnozę	
EU4	Potrafi wykorzystać algorytmy i biblioteki podprogramów do realizacji prostych aplikacji	INF_U02
EU5	Potrafi sformułować algorytm posługując się językiem programowania strukturalnego do opracowania programu komputerowego	INF_U02
Symbol efektu uczenia się	Sposoby weryfikacji efektów uczenia się	Forma zajęć, na której zachodzi weryfikacja
EU1	Kolokwium	W
EU2	Kolokwium	W
EU3	Projekt końcowy lub projekty cząstkowe na zajęciach	Ps
EU4	Projekt końcowy lub projekty cząstkowe na zajęciach	Ps
EU5	Projekt końcowy lub projekty cząstkowe na zajęciach	Ps
Bilans nakładu pracy słuchacza (w godzinach)		Liczba godz.
Wyliczenie	Udział w wykładach	16
	Udział w pracowni specjalistycznej	16
	Przygotowanie do pracowni specjalistycznej	40
	Opracowanie sprawozdań z pracowni i wykonanie zadań domowych (prac domowych)	49
	Udział w konsultacjach związanych projektami	2
	Realizacja zadań projektowych	50
	Przygotowanie do kolokwium i obecność na nim	30
	RAZEM:	203
Wskaźniki ilościowe		GODZINY ECTS
Nakład pracy słuchacza związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela		34 1,5
Nakład pracy słuchacza związany z zajęciami o charakterze praktycznym		157 6
Literatura podstawowa	1. B. Keringhan, D. Ritchie, Język ANSI C. Programowanie, Helion, 2010. 2. S. Prata, Język C. Szkoła programowania, Helion, 2006. 3. D. Harel, Y. Feldman, Rzecz o istocie informatyki. Algorytmika, WNT, 2009.	
Literatura uzupełniająca	1. H. Schildt, Leksykon C/C++ (standard ANSI/ISO), LT&P, 2002. 2. P. Stańczyk, Algorytmika praktyczna. Nie tylko dla mistrzów, PWN, 2009. 3. N. Wirth, Algorytmy + struktury danych = programy, WNT, 2004.	
Jednostka realizująca	Wydział Informatyki Politechniki Białostockiej	Data opracowania programu
Program opracowała	<i>dr inż. Dorota Duda</i>	12.04.2019

Wydział Informatyki										
Kierunek studiów	Informatyka - technologie ICT i programowanie							Poziom i forma studiów	studia podyplomowe	
Specjalność / ścieżka dyplomowania								Profil kształcenia		
Nazwa przedmiotu	Technologie sieciowe i standardy WWW							Kod przedmiotu	SPI005	
								Rodzaj przedmiotu	obowiązkowy	
Formy zajęć i liczba godzin	W	C	L	P	Ps	T	S	Semestr	1	
	16				16			Punkty ECTS	6	
Przedmioty wprowadzające										
Cele przedmiotu	Słuchacze zapoznają się z budową i działaniem sieci komputerowych. Poznają najczęściej wykorzystywane protokoły komunikacyjne, urządzenia i media stosowane w sieciach komputerowych. Będą potrafili nazwać i scharakteryzować poszczególne warstwy modelu referencyjnego oraz technologie stosowane w sieciach komputerowych. Będą również potrafili zbudować prostą sieć komputerową oraz znaleźć i naprawić podstawowe usterki w działaniu sieci.									
Treści programowe	Wykład: Podstawowe pojęcia związane z sieciami komputerowymi. Podział sieci na podsieci. Modele referencyjne ISO/OSI oraz TCP/IP. Wybrane protokoły poszczególnych warstw (m.in. DHCP, ARP, IP, TCP, UDP). Protokoły routingu (m.in. RIP, OSPF, BGP). Wykrywanie i usuwanie uszkodzeń w sieciach komputerowych. Protokół HTTP/HTTPS oraz konfiguracja serwera WWW. Kierunki rozwoju sieci komputerowych. Pracownia specjalistyczna: Przygotowanie kabla UTP, połączenie bezpośrednie urządzeń, podstawowa konfiguracja interfejsów sieciowych. Konfiguracja sieci VLAN na przełącznikach. Konfiguracja serwera DHCP. Konfiguracja maskarady. Konfiguracja serwera WWW.									
Metody dydaktyczne	1. prezentacja multimedialna; 2. analiza wybranych przypadków; 3. samodzielne opracowanie wskazanych zagadnień; 4. ćwiczenia indywidualne, ćwiczenia grupowe; 5. przygotowanie przez słuchacza samodzielnej koncepcji projektowej; 6. dyskusje dydaktyczne.									
Forma zaliczenia	Wykład – kolokwium, Pracownia specjalistyczna – sprawozdania z wykonywanych zadań praktycznych									
Symbol efektu uczenia się	Zakładane efekty uczenia się								Odniesienie do kierunkowych efektów uczenia się	
EU1	Zna podstawy budowy i działania sieci komputerowych								INF_W01	
EU2	Zna podstawowe pojęcia związane z sieciami komputerowymi								INF_W01	
EU3	Potrafi skonfigurować komputer lub urządzenie sieciowe do pracy w sieci komputerowej								INF_U01	
EU4	Potrafi samodzielnie i zespołowo konfigurować wybrane usługi sieciowe oraz dokumentować swoją pracę								INF_U01, INF_S01	
Symbol efektu uczenia się	Sposoby weryfikacji efektów uczenia się								Forma zajęć, na której zachodzi weryfikacja	
EU1	Kolokwium								W	
EU2	Kolokwium								W	

EU3	Sprawozdanie	Ps
EU4	Sprawozdanie	Ps
Bilans nakładu pracy słuchacza (w godzinach)		Liczba godz.
Wyliczenie	Udział w wykładach	16
	Udział w pracowni specjalistycznej	16
	Przygotowanie do pracowni specjalistycznej	32
	Opracowanie sprawozdań z pracowni i wykonanie zadań domowych (prac domowych)	68
	Udział w konsultacjach związanych z pracownią specjalistyczną	2
	Przygotowanie do kolokwium	20
	RAZEM:	150
Wskaźniki ilościowe		GODZINY ECTS
Nakład pracy słuchacza związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela		32 1,5
Nakład pracy słuchacza związany z zajęciami o charakterze praktycznym		116 4,5
Literatura podstawowa	1. D.E. Comer, Sieci komputerowe i intersieci: kompendium wiedzy każdego administratora, Helion, 2014. 2. A. Tanenbaum, Sieci komputerowe, Prentice Hall, Indian International Ed., 2010. 3. Dokumentacja pakietu netfilter, https://www.netfilter.org . 4. Serwer DHCP, https://www.isc.org/dhcp/ . 5. Dokumentacja pakietu Apache, https://www.apache.org .	
Literatura uzupełniająca	1. Dokumentacje techniczne dostępne w Internecie (dokumenty RFC, instrukcje do oprogramowania itp.). 2. Dokumentacja systemu Mikrotik, https://www.mikrotik.com .	
Jednostka realizująca	Wydział Informatyki Politechniki Białostockiej	Data opracowania programu
Program opracował	<i>dr inż. Andrzej Chmielewski</i> <i>dr inż. Mirosław Omieljanowicz</i>	11.04.2019

Wydział Informatyki										
Kierunek studiów	Informatyka - technologie ICT i programowanie							Poziom i forma studiów	studia podyplomowe	
Specjalność / ścieżka dyplomowania								Profil kształcenia		
Nazwa przedmiotu	Programowanie obiektowe i zdarzeniowe							Kod przedmiotu	SPI006	
								Rodzaj przedmiotu	obowiązkowy	
Formy zajęć i liczba godzin	W	C	L	P	Ps	T	S	Semestr	2	
	16				16			Punkty ECTS	7	
Przedmioty wprowadzające										
Cele przedmiotu	Zapoznanie słuchacza z podstawowymi technikami programowania obiektowego i zdarzeniowego oraz technologią WPF i ich wykorzystaniem do tworzenia aplikacji użytkowych działających w środowisku Windows w .NET Framework. Przedstawienie zasad tworzenia interfejsu użytkownika w języku XAML. Nauczenie sposobu wykorzystywania zdobytej wiedzy poprzez realizację prostych zadań na zajęciach. Realizacja prostych aplikacji w poznanej technologii.									
Treści programowe	Wykład: Wprowadzenie do programowania obiektowego i zdarzeniowego. Tworzenie interfejsu użytkownika w WPF. Pojęcie klasy i obiektu. Składniki klas – pola i metody. Konstruktory. Przegląd najważniejszych rodzajów kontrolek. Posługiwanie się oknami dialogowymi. Kompozycja. Kolekcje w języku C#. Technika wiązania danych. Dziedziczenie w programowaniu obiektowym. Polimorfizm i metody wirtualne. Mechanizm wyjątków. Modyfikowanie wyglądu aplikacji przy użyciu stylów. Umieszczanie grafiki i animacji w aplikacji WPF. Pracownia specjalistyczna: Tworzenie interfejsu użytkownika w WPF. Pojęcie klasy i obiektu. Składniki klas – pola i metody. Konstruktory. Przegląd najważniejszych rodzajów kontrolek. Posługiwanie się oknami dialogowymi. Kompozycja. Kolekcje w języku C#. Technika wiązania danych. Dziedziczenie w programowaniu obiektowym. Polimorfizm i metody wirtualne. Mechanizm wyjątków. Modyfikowanie wyglądu aplikacji przy użyciu stylów. Umieszczanie grafiki i animacji w aplikacji WPF.									
Metody dydaktyczne	1. prezentacja multimedialna; 2. analiza wybranych przypadków; 3. samodzielne opracowanie wskazanych zagadnień; 4. ćwiczenia indywidualne, ćwiczenia grupowe; 5. przygotowanie przez słuchacza samodzielnej koncepcji projektowej; 6. dyskusje dydaktyczne.									
Forma zaliczenia	Wykład – kolokwium, Pracownia specjalistyczna – projekt końcowy lub projekty częściowe na zajęciach									
Symbol efektu uczenia się	Zakładane efekty uczenia się								Odniesienie do kierunkowych efektów uczenia się	
EU1	Posiada podstawową wiedzę w zakresie metodyk i technik programowania obiektowego i zdarzeniowego								INF_W02	
EU2	Posiada wiedzę pozwalającą na tworzenie aplikacji użytkowych w .NET Framework z wykorzystaniem prostych algorytmów i struktur danych								INF_W02	
EU3	Zna język programowania obiektowego C# oraz język XAML do projektowania interfejsów użytkownika aplikacji tworzonych w								INF_U02, INF_S02	

	technologii WPF na poziomie pozwalającym na wejście na rynek pracy	
EU4	Potrafi projektować i implementować proste aplikacje z realizujące zadane algorytmy, z wykorzystaniem technologii WPF	INF_U02
EU5	Potrafi przetestować program obiektowy, kod źródłowy i zdiagnozować błędy	INF_U02
Symbol efektu uczenia się	Sposoby weryfikacji efektów uczenia się	Forma zajęć, na której zachodzi weryfikacja
EU1	Kolokwium	W
EU2	Kolokwium	W
EU3	Projekt końcowy lub projekty cząstkowe na zajęciach	Ps
EU4	Projekt końcowy lub projekty cząstkowe na zajęciach	Ps
EU5	Projekt końcowy lub projekty cząstkowe na zajęciach	Ps
Bilans nakładu pracy słuchacza (w godzinach)		Liczba godz.
Wyliczenie	Udział w wykładach	16
	Udział w pracowni specjalistycznej	16
	Przygotowanie do pracowni specjalistycznej	60
	Opracowanie sprawozdań z pracowni specjalistycznej i wykonanie zadań domowych (prac domowych)	60
	Udział w konsultacjach związanych z projektem	2
	Realizacja zadań projektowych	14
	Przygotowanie do kolokwium i obecność na nim	20
	RAZEM:	188
Wskaźniki ilościowe		GODZINY ECTS
Nakład pracy słuchacza związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela		34 1,5
Nakład pracy słuchacza związany z zajęciami o charakterze praktycznym		152 5,5
Literatura podstawowa	1. A. Stellman, C# , Helion, 2011. 2. M. MacDonald, Pro WPF in C# 2010: Windows Presentation Foundation in .NET 4, Apress, 2010.	
Literatura uzupełniająca	1. J. Matulewski [et al.], Visual Studio 2010 dla programistów C#, Helion, 2011. 2. A. Boduch, Wstęp do programowania w języku C#, Helion, 2010. 3. A. Troelsen, Język C # 2008 i platforma .NET 3.5, Wydaw. Naukowe PWN, 2009. 4. A. Nathan, WPF 4 Unleashed, SAMS, 2010.	
Jednostka realizująca	Wydział Informatyki Politechniki Białostockiej	Data opracowania programu
Program opracował	<i>dr inż. Marek Tabędzki</i>	11.04.2019

Wydział Informatyki										
Kierunek studiów	Informatyka - technologie ICT i programowanie							Poziom i forma studiów	studia podyplomowe	
Specjalność / ścieżka dyplomowania								Profil kształcenia		
Nazwa przedmiotu	Inżynieria oprogramowania i analiza biznesowa							Kod przedmiotu	SPI007	
								Rodzaj przedmiotu	obowiązkowy	
Formy zajęć i liczba godzin	W	C	L	P	Ps	T	S	Semestr	2	
	16				16			Punkty ECTS	6	
Przedmioty wprowadzające										
Cele przedmiotu	Celem wykładu jest przedstawienie całego procesu związanego z tworzeniem i wykorzystywaniem systemów informatycznych. Powinien on uświadomić słuchaczom, że programowanie jest tylko elementem składowym tego procesu oraz, że na powodzenie przedsięwzięcia mają wpływ wszystkie fazy cyklu życia oprogramowania. Celem zajęć pracowni specjalistycznej jest nauczanie modelowania i projektowania przy użyciu Unified Modeling Language (UML).									
Treści programowe	Wykład: Przedmiot i cele inżynierii oprogramowania, przyczyny powstania IO, modele cyklu życia systemów informatycznych; Analiza i projektowanie obiektowe przy wykorzystaniu Unified Modeling Language (UML): diagramy przypadków użycia, diagramy czynności, diagramy klas i obiektów, diagramy zmiany stanów, diagramy interakcji i diagramy fizyczne; Inżynieria wymagań dla systemów informatycznych; Projektowanie, architektura, implementacja, wdrażanie systemów informatycznych; Zapewnienie jakości oprogramowania i wiarygodność systemów informatycznych; Zarządzanie projektami informatycznymi. Pracownia specjalistyczna: Pokaz działania wybranych narzędzi CASE; Analiza i projektowanie obiektowe przy wykorzystaniu Unified Modeling Language (UML): diagramy przypadków użycia, diagramy czynności, diagramy klas i obiektów, diagramy zmiany stanów, diagramy interakcji i diagramy fizyczne; Inżynieria wymagań dla systemów informatycznych. Etyczne aspekty pracy.									
Metody dydaktyczne	1. prezentacja multimedialna; 2. analiza wybranych przypadków; 3. samodzielne opracowanie wskazanych zagadnień; 4. ćwiczenia indywidualne, ćwiczenia grupowe; 5. przygotowanie przez słuchacza samodzielnej koncepcji projektowej; 6. dyskusje dydaktyczne.									
Forma zaliczenia	Wykład – kolokwium, Pracownia specjalistyczna – sprawdziany, obserwacja pracy na zajęciach									
Symbol efektu uczenia się	Zakładane efekty uczenia się								Odniesienie do kierunkowych efektów uczenia się	
EU1	Zna i rozumie zasady inżynierii oprogramowania, metody i techniki wykorzystywane w projektowaniu systemów informatycznych; zna języki modelowania i komputerowe narzędzia wspomagające projektowanie								INF_W04	
EU2	Posiada elementarną wiedzę na temat cyklu życia oprogramowania; zna i rozumie procesy wytwarzania, wdrażania i utrzymania oprogramowania								INF_W04	

EU3	Posiada elementarną wiedzę w zakresie zarządzania przedsiębiorstwami informatycznymi	INF_W04	
EU4	Potrafi opracować elementy dokumentacji zadania inżynierskiego: specyfikację wymagań, model architektury, realizację fizyczną	INF_U04	
EU5	Potrafi działać zespołowo przy projektowaniu systemu komputerowego, przestrzegając zasad etyki	INF_U04, INF_S03	
Symbol efektu uczenia się	Sposoby weryfikacji efektów uczenia się	Forma zajęć, na której zachodzi weryfikacja	
EU1	Kolokwium	W	
EU2	Kolokwium	W	
EU3	Kolokwium	W	
EU4	Sprawdziany na pracowni, obserwacja pracy na zajęciach	Ps	
EU5	Sprawdziany na pracowni, obserwacja pracy na zajęciach	Ps	
Bilans nakładu pracy słuchacza (w godzinach)		Liczba godz.	
Wyliczenie	Udział w wykładach	16	
	Udział w pracowni specjalistycznej	16	
	Przygotowanie do zajęć pracowni specjalistycznej	32	
	Opracowanie sprawozdań z pracowni specjalistycznej i wykonanie zadań domowych (prac domowych)	64	
	Udział w konsultacjach	2	
	Przygotowanie do kolokwium i obecność na nim	22	
	RAZEM:	152	
Wskaźniki ilościowe		GODZINY	ECTS
Nakład pracy słuchacza związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela		34	1,5
Nakład pracy słuchacza związany z zajęciami o charakterze praktycznym		114	4,5
Literatura podstawowa	1. K. Subieta, Wprowadzenie do inżynierii oprogramowania, Wydawnictwo PJWSTK, 2002. 2. K. Sacha, Inżynieria oprogramowania, PWN, 2010. 3. A. Koszłajda, Zarządzanie projektami IT. Przewodnik po metodykach, Helion, 2010. 4. D. Pilone, N. Pitman, UML 2.0. Almanach, Helion, O'Reilly, 2007.		
Literatura uzupełniająca	1. M. E. Bays, Metodyka wprowadzania oprogramowania na rynek, WNT, 2001. 2. P. Graessle, H. Baumann, P. Baumann, UML 2.0 w akcji. Przewodnik oparty na projektach, Helion, 2006. 3. I. Sommerville, Software engineering, Pearson Education, 2004.		
Jednostka realizująca	Wydział Informatyki Politechniki Białostockiej	Data opracowania programu	
Program opracował	<i>dr hab. inż. Marek Krętowski, prof. PB</i>	11.04.2019	

Wydział Informatyki										
Kierunek studiów	Informatyka - technologie ICT i programowanie							Poziom i forma studiów	studia podyplomowe	
Specjalność / ścieżka dyplomowania								Profil kształcenia		
Nazwa przedmiotu	Programowanie aplikacji internetowych							Kod przedmiotu	SPI008	
								Rodzaj przedmiotu	obowiązkowy	
Formy zajęć i liczba godzin	W	C	L	P	Ps	T	S	Semestr	2	
	16				16			Punkty ECTS	6	
Przedmioty wprowadzające										
Cele przedmiotu	Zapoznanie słuchaczy z podstawami tworzenia statycznych stron WWW oraz dynamicznych aplikacji internetowych w technologii .NET. Słuchacze nauczą się HTML i CSS, ASP.NET oraz technologii jQuery. Wszystkie te elementy pozwolą na tworzenie aplikacji internetowych.									
Treści programowe	Wprowadzenie do języka HTML. Podstawy stylów CSS. Środowisko Visual Studio 2013. Kontrolki serwerowe i walidacyjne w technologii ASP .NET. Technologia jQuery. Technologia AJAX. Dostęp do baz danych z wykorzystaniem biblioteki ADO. Kontrolki do przetwarzania dużych ilości danych. Kontrolki użytkownika. Mechanizm MasterPage.									
Metody dydaktyczne	1. prezentacja multimedialna; 2. analiza wybranych przypadków; 3. samodzielne opracowanie wskazanych zagadnień; 4. ćwiczenia indywidualne, ćwiczenia grupowe; 5. przygotowanie przez słuchacza samodzielnej koncepcji projektowej; 6. dyskusje dydaktyczne.									
Forma zaliczenia	Wykład – kolokwium, Pracownia specjalistyczna - prace domowe, obserwacja pracy na zajęciach, wykonany i działający program komputerowy									
Symbol efektu uczenia się	Zakładane efekty uczenia się								Odniesienie do kierunkowych efektów uczenia się	
EU1	Zna podstawy HTML								INF_W05	
EU2	Zna podstawy CSS								INF_W05	
EU3	Rozumie podstawowe elementy technologii jQuery.								INF_U05	
EU4	Zna i opisuje przeznaczenie tagów HTML oraz składni języka CSS								INF_U05	
EU5	Potrafi stworzyć wykorzystać elementy ASP.NET do stworzenia aplikacji internetowej								INF_U05	
Symbol efektu uczenia się	Sposoby weryfikacji efektów uczenia się								Forma zajęć, na której zachodzi weryfikacja	
EU1	Kolokwium								W	
EU2	Kolokwium								W	
EU3	Praca domowa, obserwacja pracy na zajęciach								Ps	
EU4	Praca domowa, wykonany i działający program komputerowy								Ps	
EU5	Praca domowa, obserwacja pracy na zajęciach								Ps	
Bilans nakładu pracy słuchacza (w godzinach)									Liczba godz.	

Wyliczenie	Udział w wykładach	16	
	Udział w pracowni specjalistycznej	16	
	Przygotowanie do pracowni specjalistycznej	40	
	Wykonanie zadań domowych z pracowni specjalistycznej	40	
	Udział w konsultacjach	2	
	Przygotowanie projektu semestralnego	40	
	RAZEM:	158	
Wskaźniki ilościowe		GODZINY	ECTS
Nakład pracy słuchacza związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela		34	1,5
Nakład pracy słuchacza związany z zajęciami o charakterze praktycznym		138	5,5
Literatura podstawowa	1. D. Esposito, Microsoft ASP.NET and AJAX: architecting Web applications, Microsoft Press, 2009. 2. Materiały szkoleniowe z kursu Microsoft IT Academy "Aplikacje Internetowe". 3. B. Pfaffenberger, S. M. Schafer, C. White, B. Karow, HTML, XHTML i CSS. Biblia, Helion, 2005.		
Literatura uzupełniająca	1. G. Shepherd, Microsoft ASP.NET 2.0 Step By Step, Microsoft Press, 2005. 2. Douglas J. Reilly, Programowanie Web Forms w ASP.NET 2.0+, APN Promise, 2006. 3. B. Lawson, R. Sharp, Wprowadzenie do HTML5. Autorytety Informatyki, Helion, 2011		
Jednostka realizująca	Wydział Informatyki Politechniki Białostockiej	Data opracowania programu	
Program opracował	<i>dr inż. Paweł Tadejko, dr inż. Maciej Kopczyński</i>	11.04.2019	

Wydział Informatyki										
Kierunek studiów	Informatyka - technologie ICT i programowanie							Poziom i forma studiów	studia podyplomowe	
Specjalność / ścieżka dyplomowania								Profil kształcenia		
Nazwa przedmiotu	Bezpieczeństwo systemów informatycznych							Kod przedmiotu	SPI009	
								Rodzaj przedmiotu	obowiązkowy	
Formy zajęć i liczba godzin	W	C	L	P	Ps	T	S	Semestr	2	
	10				10			Punkty ECTS	4	
Przedmioty wprowadzające										
Cele przedmiotu	Przedmiot jest wprowadzeniem do kryptografii oraz zagadnień bezpieczeństwa systemów informatycznych. Słuchacze poznają podstawowe pojęcia związane z usługami bezpieczeństwa w systemach operacyjnych i sieciach komputerowych, dowiedzą się również na jakie elementy należy zwrócić uwagę przy ocenie stopnia zabezpieczeń. W ramach kursu zaprezentowane zostaną standardy zabezpieczeń oraz ich wykorzystanie w usługach elektronicznych świadczonych przez organy administracji państwowej.									
Treści programowe	Wykład: 1. Przed czym mamy się bronić - rodzaje zagrożeń w systemach teleinformatycznych. 2. Anonimowość w sieci. 3. Podstawy kryptografii. 4. Wybrane ataki na sieci komputerowe. 5. Usługi podnoszące poziom bezpieczeństwa. Pracownia specjalistyczna: 1. Konfiguracja usług sieciowych podnoszących poziom bezpieczeństwa (SSH, HTTPS). 2. Konfiguracja firewalla. 3. Przeprowadzenie symulacji wybranego ataku.									
Metody dydaktyczne	1. prezentacja multimedialna; 2. analiza wybranych przypadków; 3. samodzielne opracowanie wskazanych zagadnień; 4. ćwiczenia indywidualne, ćwiczenia grupowe.									
Forma zaliczenia	Wykład – kolokwium, Pracownia specjalistyczna - wykonanie zadań na zajęciach									
Symbol efektu uczenia się	Zakładane efekty uczenia się								Odniesienie do kierunkowych efektów uczenia się	
EU1	Zna podstawowe pojęcia związane z usługami bezpieczeństwa takie jak: weryfikacja tożsamości, autoryzacja, poufność, integralność oraz niezaprzeczalność								INF_W01	
EU2	Zna podstawowe typy ataków na systemy komputerowe								INF_W01	
EU3	Potrafi konfigurować wybrane usługi sieciowe zorientowane na bezpieczeństwo								INF_U01	
EU4	Potrafi przeprowadzić symulację wybranego ataku								INF_U01	
Symbol efektu uczenia się	Sposoby weryfikacji efektów uczenia się								Forma zajęć, na której zachodzi weryfikacja	

EU1	Kolokwium	W	
EU2	Kolokwium	W	
EU3	Wykonanie zadań na zajęciach	Ps	
EU4	Wykonanie zadań na zajęciach	Ps	
Bilans nakładu pracy słuchacza (w godzinach)		Liczba godz.	
Wyliczenie	Udział w wykładach	10	
	Udział w pracowni specjalistycznej	10	
	Przygotowanie do pracowni specjalistycznej	20	
	Opracowanie sprawozdań z pracowni specjalistycznej i wykonanie zadań domowych (prac domowych)	58	
	Udział w konsultacjach	2	
	RAZEM:	100	
Wskaźniki ilościowe		GODZINY	ECTS
Nakład pracy słuchacza związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela		22	1
Nakład pracy słuchacza związany z zajęciami o charakterze praktycznym		90	3,5
Literatura podstawowa	1. J.P. Aumasson, Nowoczesna kryptografia: praktyczne wprowadzenie do szyfrowania, PWN, 2018. 2. W. Stallings , L. Brown, Bezpieczeństwo systemów informatycznych. Zasady i praktyka, Helion, 2019. 3. J. Muniz, A. Lakhani, Kali Linux. Testy penetracyjne, Helion, 2014. 4. C. Binnie, Linux Server: bezpieczeństwo i ochrona sieci, Helion, 2017.		
Literatura uzupełniająca	1. S. Suehring, Zapory sieciowe w systemie Linux: kompendium wiedzy o nftables, Helion, 2015. 2. K. Mitnick, Sztuka podstępów: łamałem ludzi, nie hasła, Helion, 2016. 3. Netfilter, https://www.netfilter.org 4. Apache project, https://www.apache.org 5. OpenSSH, https://openssh.org 6. OpenVPN, https://openvpn.net .		
Jednostka realizująca	Wydział Informatyki Politechniki Białostockiej	Data opracowania programu	
Program opracował	<i>dr inż. Andrzej Chmielewski</i>	11.04.2019	

Wydział Informatyki										
Kierunek studiów	Informatyka - technologie ICT i programowanie							Poziom i forma studiów	studia podyplomowe	
Specjalność / ścieżka dyplomowania								Profil kształcenia		
Nazwa przedmiotu	Projektowanie UX i interfejsów użytkownika							Kod przedmiotu	SPI010	
								Rodzaj przedmiotu	obowiązkowy	
Formy zajęć i liczba godzin	W	C	L	P	Ps	T	S	Semestr	2	
	8				8			Punkty ECTS	5	
Przedmioty wprowadzające										
Cele przedmiotu	Celem przedmiotu jest zapoznanie słuchaczy z zagadnieniami przygotowania architektury informacji rozwiązań internetowych zgodnie z najnowszymi trendami UI/UX oraz ich dalszej optymalizacji. Przedmiot ma również na celu omówienie podstaw nowoczesnego web designu. Słuchacze przygotowują założenia i wymagania, a następnie prototypu wireframes serwisu internetowego.									
Treści programowe	Wykład: UI/UX w projektowaniu serwisów internetowych. Tworzenie prototypów webowych i mobilnych. Podstawowe zasady nowoczesnego projektowania graficznego. Analityka webowa w ocenie skuteczności przygotowanych rozwiązań. Zasady skutecznej optymalizacji serwisów internetowych. Pracownia specjalistyczna: Opracowanie założeń i wymagań dla projektu internetowego. Opracowanie prototypu wireframes serwisu internetowego.									
Metody dydaktyczne	1. prezentacja multimedialna; 2. analiza wybranych przypadków; 3. samodzielne opracowanie wskazanych zagadnień; 4. ćwiczenia indywidualne, ćwiczenia grupowe; 5. przygotowanie przez słuchacza samodzielnej koncepcji projektowej; 6. dyskusje dydaktyczne.									
Forma zaliczenia	Wykład – kolokwium, Pracownia specjalistyczna – dokument z opisem wymagań i założeń projektu internetowego, prototyp wireframes serwisu internetowego, dokumentacja projektu									
Symbol efektu uczenia się	Zakładane efekty uczenia się								Odniesienie do kierunkowych efektów uczenia się	
EU1	Zna podstawowe zasady Usability, User Experience i Web designu								INF_W03	
EU2	Zna najważniejsze metody optymalizacji projektów internetowych								INF_W03	
EU3	Potrafi przygotować założenia i wymagania dla planowanego rozwiązania internetowego								INF_U03	
EU4	Potrafi przygotować prototypu wireframes serwisu internetowego								INF_U03	
Symbol efektu uczenia się	Sposoby weryfikacji efektów uczenia się								Forma zajęć, na której zachodzi weryfikacja	
EU1	Kolokwium								W	
EU2	Kolokwium								W	
EU3	Dokumentacja projektu końcowego, dokument z opisem								Ps	

	wymagań i założeń planowanego projektu internetowego	
EU4	Prototyp wireframes serwisu internetowego	Ps
Bilans nakładu pracy słuchacza (w godzinach)		Liczba godz.
Wyliczenie	Udział w wykładach	8
	Udział w pracowni specjalistycznej	8
	Przygotowanie do pracowni specjalistycznej	40
	Wykonanie zadań domowych z pracowni specjalistycznej	30
	Udział w konsultacjach	2
	Realizacja zadań projektowych	12
	Przygotowanie projektu semestralnego	35
	Przygotowanie do kolokwium i obecność na nim	5
	Przygotowanie do zaliczenia pracowni specjalistycznej i obecność na nim	5
	RAZEM:	149
Wskaźniki ilościowe		GODZINY ECTS
Nakład pracy słuchacza związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela		18 0,5
Nakład pracy słuchacza związany z zajęciami o charakterze praktycznym		132 4,5
Literatura podstawowa	1. S. Krug, Nie każ mi myśleć! O życiowym podejściu do funkcjonalności stron internetowych, Helion, 2006 2. J. Beaird, Niezawodne zasady web designu. Projektowanie spektakularnych witryn internetowych, Helion, 2012 3. B. Eisenberg, J. Quarto-von Tivadar, B. Crosby, L. T. David, Google Website Optimizer. Przewodnik, Helion, 2010 4. R. K. Wysocki, Efektywne zarządzanie projektami, Helion, 2013.	
Literatura uzupełniająca	1. http://uxmag.com 2. http://uxbite.com 3. http://online-behavior.com 4. http://www.getelastic.com 5. http://blog.kissmetrics.com	
Jednostka realizująca	Wydział Informatyki Politechniki Białostockiej	Data opracowania programu
Program opracowała	<i>Magdalena Jagodziki</i>	11.04.2019

Wydział Informatyki										
Kierunek studiów	Informatyka - technologie ICT i programowanie							Poziom i forma studiów	studia podyplomowe	
Specjalność / ścieżka dyplomowania								Profil kształcenia		
Nazwa przedmiotu	Techniki zapewniania jakości oprogramowania							Kod przedmiotu	SPI012	
								Rodzaj przedmiotu	obowiązkowy	
Formy zajęć i liczba godzin	W	C	L	P	Ps	T	S	Semestr	1	
	8							Punkty ECTS	1	
Przedmioty wprowadzające										
Cele przedmiotu	Celem przedmiotu jest zaznajomienie słuchaczy ze sposobami pisania prawidłowego i bezpiecznego oprogramowania jak również z metodykami zarządzania błędami w systemach informatycznych.									
Treści programowe	Wykład: 1. Zasady tworzenia czystego kodu w programowaniu obiektowym. 2. Testowanie funkcjonalne, strukturalne. 3. Metody pisania bezpiecznego oprogramowania. 4. Narzędzia procesu tworzenia oprogramowania. 5. Metodyki SOLID, KISS, DRY. 6. Cykl życia produktu a cykl testowania. 7. Metody zarządzania błędami oprogramowania. Procedury naprawcze.									
Metody dydaktyczne	1. Prezentacja multimedialna 2. Wykład problemowy 3. Wykład informacyjny									
Forma zaliczenia	Wykład: kolokwium zaliczeniowe									
Symbol efektu uczenia się	Zakładane efekty uczenia się								Odniesienie do kierunkowych efektów uczenia się	
EU1	Zna podstawowe definicje i metody związane z tworzeniem bezpiecznego oprogramowania wysokiej jakości								INF_W02, INF_W04	
EU2	Zna zasady analizy programów i wyszukiwania fragmentów kodu niskiej jakości oraz zagrażających bezpieczeństwu systemu informatycznego								INF_W02, INF_W04	
EU3	Zna zasady porównywania metodyki zapewniania jakości oprogramowania								INF_W04	
EU4	Zna narzędzia procesu tworzenia oprogramowania i zasady tworzenia czystego kodu								INF_W02, INF_W04	
EU5	Rozumie czym jest cykl życia produktu i cykl testowania oraz potrafi zaplanować przebieg testów oprogramowania								INF_W04	
Symbol efektu uczenia się	Sposoby weryfikacji efektów uczenia się								Forma zajęć, na której zachodzi	

		weryfikacja	
EU1	Kolokwium zaliczeniowe	W	
EU2	Kolokwium zaliczeniowe	W	
EU3	Kolokwium zaliczeniowe	W	
EU4	Kolokwium zaliczeniowe	W	
EU5	Kolokwium zaliczeniowe	W	
Bilans nakładu pracy słuchacza (w godzinach)		Liczba godz.	
Wyliczenie	Udział w wykładach	8	
	Udział w konsultacjach związanych z wykładem	2	
	Przygotowanie do kolokwium zaliczeniowego	6	
	Uczestnictwo w kolokwium zaliczeniowym	2	
	Analiza we własnym zakresie programów udostępnionych po każdym wykładzie.	8	
	RAZEM:	26	
Wskaźniki ilościowe		GODZINY	ECTS
Nakład pracy słuchacza związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela		12	0,5
Nakład pracy słuchacza związany z zajęciami o charakterze praktycznym		0	0
Literatura podstawowa	1. R. C. Martin, Czysty kod. Podręcznik dobrego programisty, Helion, 2014. 2. A. Roman, Testowanie i jakość oprogramowania. Modele, techniki, narzędzia, Wydawnictwo Naukowe PWN, 2017. 3. A. Roman, K. Zmitrowicz, Testowanie oprogramowania w praktyce. Studium przypadków, Wydawnictwo Naukowe PWN, 2016.		
Literatura uzupełniająca	1. K. Zmitrowicz, Jakość projektów informatycznych. Rozwój i testowanie oprogramowania, Wydawnictwo Naukowe PWN, 2015. 2. R. Rajani, Testowanie kodu w praktyce, Helion, 2018. 3. G. Coldwind, Zrozumieć programowanie, Wydawnictwo Naukowe PWN, 2016. 4. S. McConnell, Code complete: a practical handbook of software construction, Microsoft Press, 1993.		
Jednostka realizująca	Wydział Informatyki Politechniki Białostockiej	Data opracowania programu	
Program opracował	<i>mgr inż. Maciej Szymkowski</i>	18.04.2019	

Wydział Informatyki										
Kierunek studiów	Informatyka - technologie ICT i programowanie							Poziom i forma studiów	studia podyplomowe	
Specjalność / ścieżka dyplomowania								Profil kształcenia		
Nazwa przedmiotu	Wprowadzenie do Biometrii							Kod przedmiotu	SPI014	
								Rodzaj przedmiotu	obowiązkowy	
Formy zajęć i liczba godzin	W	C	L	P	Ps	T	S	Semestr	2	
	8				10			Punkty ECTS	2	
Przedmioty wprowadzające										
Cele przedmiotu	Celem przedmiotu jest wprowadzenie słuchaczy do zagadnień związanych z rozpoznawaniem człowieka na podstawie cech biometrycznych. Słuchacze poznają podstawowe definicje i pojęcia dotyczące biometrii, cechy biometryczne i ich klasyfikację. Słuchacze nauczą się implementować w języku Python proste algorytmy operujące na obrazach i sygnałach oraz używać je do konstrukcji systemu biometrycznego.									
Treści programowe	Wykład: Podstawowe pojęcia biometryczne. Klasyfikacje cech biometrycznych. Biometria behawioralna, fizjologiczna i hybrydowa. Urządzenia i sensory biometryczne. Metody rozpoznawania człowieka na podstawie: twarzy, odcisków palców, tęczy oka, siatkówki oka. Wprowadzenie do przetwarzania obrazów i sygnałów. Pracownia specjalistyczna: Budowa i wykorzystanie urządzeń i sensorów w biometrii. Algorytmy przetwarzania i analizy obrazów oraz sygnałów. Rozpoznawanie człowieka na podstawie twarzy oraz głosu.									
Metody dydaktyczne	1. prezentacja multimedialna; 2. wykład problemowy; 3. wykład informacyjny; 4. samodzielne opracowanie wskazanych zagadnień; 5. programowanie z użyciem komputera.									
Forma zaliczenia	Wykład – kolokwium, Pracownia specjalistyczna – projekt końcowy									
Symbol efektu uczenia się	Zakładane efekty uczenia się								Odniesienie do kierunkowych efektów uczenia się	
EU1	Zna podstawowe definicje i pojęcia dotyczące biometrii i jej rozwiązań								INF_W01	
EU2	Zna podstawowe cechy biometryczne i potrafi je sklasyfikować oraz uzasadnić możliwość ich użycia w realnych systemach biometrycznych								INF_W01	
EU3	Potrafi zaimplementować w języku Python proste algorytmy operujące na obrazach i sygnałach oraz użyć je do konstrukcji systemu biometrycznego								INF_U02, INF_U06	
EU4	Potrafi opisać schemat systemu biometrycznego i uzasadnić poprawność jego budowy								INF_U06	
Symbol efektu uczenia się	Sposoby weryfikacji efektów uczenia się								Forma zajęć, na której zachodzi weryfikacja	
EU1	Kolokwium zaliczeniowe								W	
EU2	Kolokwium zaliczeniowe								W	

EU3	Projekt końcowy	Ps	
EU4	Projekt końcowy	Ps	
Bilans nakładu pracy słuchacza (w godzinach)		Liczba godz.	
Wyliczenie	Udział w wykładach	8	
	Udział w zajęciach pracowni specjalistycznej	10	
	Przygotowanie do pracowni specjalistycznej	10	
	Wykonanie zadań domowych z pracowni specjalistycznej	10	
	Udział w konsultacjach związanych z wykładem lub pracownią specjalistyczną	2	
	Przygotowanie projektu końcowego	6	
	Przygotowanie do kolokwium zaliczeniowego	6	
	RAZEM:	52	
Wskaźniki ilościowe		GODZINY	ECTS
Nakład pracy słuchacza związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela		20	1
Nakład pracy słuchacza związany z zajęciami o charakterze praktycznym		38	1,5
Literatura podstawowa	1. R. Bolle et. al., Biometria, Wydawnictwa Naukowo-Techniczne, 2008. 2. K. Ślot, Wybrane zagadnienia biometrii, Wydawnictwa Komunikacji i Łączności, 2008. 3. K. Ślot, Rozpoznawanie biometryczne: nowe metody ilościowej reprezentacji obiektów, Wydawnictwa Komunikacji i Łączności, 2010. 4. M. Fairhurst, Biometrics: A Very Short Introduction, Oxford Press, 2019.		
Literatura uzupełniająca	1. K. Saeed et. al, New Directions in Behavioral Biometrics, CRC Press, 2017. 2. K. Saeed, T. Nagashima, Biometrics and Kansei Engineering, Springer, 2012. 3. R. Tadeusiewicz, A. Izworski, J. Majewski, Biometria, Wydawnictwa AGH Kraków, 1993. 4. A. Jain, K. Nandakumar, A. Ross, Introduction to Biometrics, Springer, 2011.		
Jednostka realizująca	Wydział Informatyki Politechniki Białostockiej	Data opracowania programu	
Program opracował	<i>mgr inż. Maciej Szymkowski</i>	11.04.2019	