

UCHWAŁA NR 18

Senatu Zachodniopomorskiego Uniwersytetu Technologicznego w Szczecinie z dnia 27 stycznia 2020 r.

w sprawie ustalenia programu studiów podyplomowych pn. Programowanie z elementami systemów wbudowanych

Na podstawie art. 28 ust. 1 pkt 11 ustawy z dnia 20 lipca 2018 r. Prawo o szkolnictwie wyższym i nauce (Dz. U. poz. 1668, z późn. zm.), Senat Zachodniopomorskiego Uniwersytetu Technologicznego w Szczecinie uchwala, co następuje:

§ 1.

Ustala się program studiów podyplomowych pn. Programowanie z elementami systemów wbudowanych. Program stanowi odpowiednio załączniki:

1. Zasady i tryb naboru na studia podyplomowe – załącznik nr 1,
2. Efekty uczenia się dla programu studiów podyplomowych – załącznik nr 2,
3. Plan studiów podyplomowych – załącznik nr 3,
4. Opisy poszczególnych przedmiotów (sylabusy) – załączniki nr 4.1-4.11,
5. Opis zasad stosowania i funkcjonowania wewnętrznego systemu zapewniania jakości kształcenia – załącznik nr 5.

§ 2.

Program studiów podyplomowych obowiązuje od semestru letniego roku akademickiego 2019/2020.

§ 3.

Traci moc Uchwała nr 111 Senatu ZUT z dnia 23 września 2019 r. w sprawie ustalenia programu studiów podyplomowych pn. Programowanie z elementami systemów wbudowanych.

§ 4.

Uchwała wchodzi w życie z dniem podjęcia.

Przewodniczący Senatu
Rektor


dr hab. inż. Jacek Wróbel, prof. ZUT

ZASADY I TRYB NABORU NA STUDIA PODYPLOMOWE

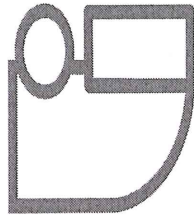
Programowanie z elementami systemów wbudowanych

(nazwa studiów)

1. Środowisko, do którego skierowana jest oferta studiów podyplomowych:
Oferta studiów podyplomowych skierowana jest do absolwentów studiów technicznych chcących zdobyć lub poszerzyć swoją wiedzę w zakresie programowania systemów wbudowanych.
2. Wymagania stawiane kandydatom:
 - absolwent wyższej uczelni,
 - biegła znajomość obsługi komputera - dostęp do komputera w domu lub pracy.
3. Kolejność przyjęć na studia podyplomowe:
O przyjęciu decyduje rozmowa kwalifikacyjna przed komisją rekrutacyjną, i spełnienie warunków przyjęcia (dokumenty oraz opłaty).
4. Wymagane dokumenty określone w §. 3 ust. 2 oraz § 3 ust. 10-12 Regulaminu studiów podyplomowych w ZUT.
5. Sposób weryfikowania i dokumentacji efektów uczenia się (gdy podstawą rekrutacji są potwierdzone efekty uczenia się)

.....
.....
.....

.....
podpis dziekana



Wydział Informatyki

Nazwa studiów: Programowanie z elementami systemów wbudowanych

Obszar/obszary kształcenia: nauki techniczne

Profil kształcenia: -

Forma studiów: niestacjonarna

Poziom kształcenia: podyplomowy

Tytuł zawodowy absolwenta: -

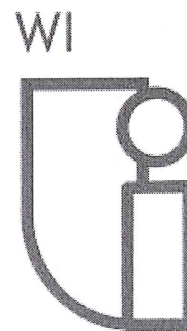
Załącznik nr 2 do uchwały nr 18 Senatu ZUT z dnia 27 stycznia 2020 r.

Kod	Efekt kształcenia dla programu kształcenia	Odniesienie do efektów kształcenia dla obszaru/obszarów kształcenia	Odniesienie do efektów uczenia się dla kwalifikacji na poziomie 6 lub 7 umożliwiających uzyskanie kompetencji inżynierskich (w przypadku studiów kończących się tytułem zawodowym inżyniera lub magistra inżyniera)	Kody treści programowych prowadzących do uzyskania efektu kształcenia
Wiedza				
PESWa_10_ W01	Sluchacz wie czym jest Linux i jakie ma zastosowanie, zna pliki i polecenia do konfiguracji systemu.	P6S_WG_TP11		T-W-1 T-W-2
PESWa_10_ W02	Sluchacz płynnie się porusza po systemie Linux i nim zarządza.	P6S_WG_TP11		T-W-3 T-W-4
PESWa_10_ W03	Sluchacz zna podstawowe narzędzia programistyczne.	P6S_WG_TP11		T-P-1 T-P-2
PESWa_10_ W04	Sluchacz wie jak implementować i wdrażać aplikacje na platformy dedykowane.	P7S_WG_TP21		T-P-1 T-P-2 T-W-1 T-W-2 T-W-3 T-W-4
PESWa_10_ W05	Sluchacz zna mechanizmy dynamicznego zarządzania pamięcią.	P7S_WG_TP21		T-L-1 T-L-2 T-L-3 T-L-4 T-P-1 T-P-2
PESWa_10_ W06	Sluchacz zna struktury danych i podstawowe algorytmy stosowane w informatyce.	P6S_WG_TP11		T-W-1 T-W-2 T-W-3
PESWa_10_ W07	Sluchacz wie jak wykorzystywać mechanizmy programistyczne udostępnione w systemie Linux.	P6S_WG_TP11		T-L-1 T-L-2 T-L-3 T-L-5 T-P-1 T-P-2
PESWa_10_ W08	Sluchacz zna metody tworzenia procesów i wątków oraz komunikacji pomiędzy nimi w systemie Linux.	P6S_WG_TP11		T-P-1 T-P-2
PESWa_10_ W09	Sluchacz zna techniki programowania proceduralnego w języku C++, zna składnię, semantykę i sferę zastosowań języka C++.	P6S_WG_TP11		T-L-1 T-L-2 T-L-3 T-P-1 T-P-2 T-W-3
PESWa_10_ W10	Sluchacz rozumie proces kompilacji oraz uruchamiania programów.	P6S_WG_TP11		T-P-1 T-P-2 T-W-1 T-W-2
PESWa_10_ W11	Sluchacz zna paradygmaty programowania obiektowego i generycznego w kontekście języka C++.	P7S_WG_TP21		T-L-1 T-L-2 T-L-3 T-P-1 T-P-2 T-W-3
PESWa_10_ W12	Sluchacz zna budowę programów wielowątkowych w języku C++.	P7S_WG_TP21		T-W-4 T-W-5 T-W-6
PESWa_10_ W13	Sluchacz zna kontenery i algorytmy biblioteki standardowej C++.	P7S_WG_TP21		T-P-1 T-P-2
PESWa_10_ W14	Student zna techniki testowania, rodzaje testów i metodykę testowania.	P6S_WG_TP11		T-L-1 T-L-2 T-L-3 T-L-4 T-L-5 T-P-1 T-P-2
PESWa_10_ W15	Sluchacz zna architekturę mikrokomputerów	P6S_WG_TP11		P-2 T-W-1 T-W-2 T-W-3 T-W-4 T-W-5

PESWa_10-_W16	Sluchacz potrafi wybrac odpowiednią metodę komunikacji z urządzeniami zewnętrznymi	P7S_WG_TP21	T-L-1 T-L-2 T-L-3 T-L-4 T-W-1 T-W-2
Umiejętności			
PESWa_10-_U01	Sluchacz potrafi obsługiwać system Linux, dokonywać jego personalizacji w zależności od zapotrzebowania.	P6S_UW_TP11	T-L-1 T-L-2 T-L-3 T-L-4
PESWa_10-_U02	Sluchacz umie implementować i instalować aplikacje na platformie dedykowanej.	P6S_UW_TP14	T-P-1 T-P-2
PESWa_10-_U03	Sluchacz potrafi wykorzystać mechanizmy dynamicznego zarządzania pamięcią.	P6S_UW_TP15	T-L-4 T-P-1 T-P-2 T-W-4
PESWa_10-_U04	Sluchacz umie zaimplementować algorytmy i struktury danych odpowiednio do zadania.	P6S_UW_TP15	T-L-1 T-L-2 T-L-3 T-L-5 T-P-1 T-P-2 T-W-1 T-W-4
PESWa_10-_U05	Sluchacz umie zaimplementować komunikację między procesami i wątkami w systemie Linux.	P6S_UW_TP15	T-L-3 T-L-4 T-P-1 T-P-2 T-W-2
PESWa_10-_U06	Sluchacz umie sterować procesem kompilacji.	P6S_UW_TP15	T-L-4 T-P-1 T-P-2 T-W-2
PESWa_10-_U07	Sluchacz umie wykorzystać własności programowania obiektowego w języku C++.	P6S_UW_TP14	T-L-2 T-L-3 T-L-4 T-P-1 T-P-2 T-W-2 T-W-3 T-W-4 T-W-5 T-W-6
PESWa_10-_U08	Sluchacz potrafi zrównoleglić zadany algorytm wykorzystując mechanizm programowania wielowątkowego w języku C++.	P6S_UW_TP14	T-P-1 T-P-2 T-W-2 T-W-3 T-W-4 T-W-5 T-W-6
PESWa_10-_U09	Sluchacz umie dobrać odpowiedni kontener do przechowania danych, oraz algorytm ze względu na wydajność.	P6S_UW_TP13	T-L-4 T-W-2 T-W-3 T-W-4 T-W-5
PESWa_10-_U10	Sluchacz umie zaprojektować i stworzyć przypadki testowe.	P6S_UW_TP14	T-L-4 T-P-1 T-P-2 T-W-6 T-W-7
PESWa_10-_U11	Sluchacz potrafi wykorzystać systemy wersjonowania plików.	P6S_UO	W-5 T-W-6 T-W-7
PESWa_10-_U12	Sluchacz umie stosować odpowiednie mechanizmy systemu Linux w zależności od potrzeb aplikacji.	P6S_UW_TP13	T-L-1 T-L-2 T-L-3 T-L-4 T-L-5 T-L-6 T-W-1 T-W-2
PESWa_10-_U13	Sluchacz potrafi zastosować wybraną metodę komunikacji z urządzeniem peryferyjnym	P6S_UW_TP14	T-L-1 T-L-2 T-L-3 T-L-4 T-W-1 T-W-2
Kompetencje społeczne			
PESWa_10-_K01	Rozumie rolę programisty w procesie wytwarzania oprogramowania.	P6S_KR	T-P-1 T-P-2
PESWa_10-_K02	Rozumie rolę testera w procesie wytwarzania oprogramowania.	P6S_KR	T-P-1 T-P-2 T-W-1 T-W-2



Wydział Informatyki



Kierunek studiów	Programowanie z elementami systemów wbudowanych					
Forma studiów	niestacjonarna	Poziom	podyplomowy			
Obszary studiów	nauki techniczne					
Profil	-					
Moduł						
Przedmiot	Wprowadzenie do systemu Linux					
Kod	/SP/PESW/01					
Jednostka prowadząca	Katedra Architektury Komputerów i Teleinformatyki					
ECTS	2,0	ECTS (formy)	2,0			
Forma zaliczenia	zaliczenie	Język	polski			
Forma dydaktyczna	Kod	Semestr	Godziny	ECTS	Waga	Zaliczenie
wykłady	W	1	4	0,7	0,50	zaliczenie
laboratoria	L	1	8	1,3	0,50	zaliczenie
Nauczyciel odpowiedzialny	Makles Krzysztof (Krzysztof.Makles@zut.edu.pl)					
Inni nauczyciele	Makles Krzysztof (Krzysztof.Makles@zut.edu.pl)					
Wymagania wstępne						
W-1	Słuchacz potrafi biegle posługiwać komputerem z dowolnym systemem operacyjnym.					
Cele modułu/przedmiotu						
C-1	Słuchacz potrafi płynnie poruszać się po systemie Linux.					
C-2	Słuchacz potrafi automatyzować podstawowe czynności w wierszu poleceń przy pomocy skryptów.					
Treści programowe z podziałem na formy zajęć						Liczba godzin
T-W-1	Wyjaśnienie czym jest Linux, jego przeznaczenie i zastosowanie. Dystrybucje i pakiety instalacyjne.					1
T-W-2	Powłoka, system plików i uprawnienia.					1
T-W-3	Poruszanie się w systemie, zarządzanie pakietami, personalizacja powłoki.					1
T-W-4	Administracja i zarządzanie systemem, przeszukiwanie systemu, podstawy Bash.					1
T-L-1	Instalacja systemu w środowisku wirtualnym					2
T-L-2	Personalizacja systemu i administracja pakietami					2
T-L-3	Przeszukiwanie systemu i administracja					2
T-L-4	Automatyzacja powtarzalnych operacji skryptami					2
Obciążenie pracą studenta - formy aktywności						Liczba godzin
A-W-1	Udział w wykładach					4
A-W-2	Samodzielnie studiowanie zalecanej literatury i analiza materiału z wykładów					15
A-W-3	Konsultacje					2
A-L-1	Uczestnictwo w zajęciach					8
A-L-2	Przygotowanie do zajęć					12
A-L-3	Samodzielne rozwiązywania zadań i problemów postawionych na laboratoriach					20
Metody nauczania / narzędzia dydaktyczne						
M-1	Wykład problemowy					
M-2	Wykład informacyjny					
M-3	Anegdota					
M-4	Ćwiczenia laboratoryjne					
M-5	Nauka i ćwiczenia własne					
Sposoby oceny (F - formująca, P - podsumowująca)						
S-1	F	Obserwacja aktywności na zajęciach laboratoryjnych				
S-2	F	Ocena kompletności rozwiązań problemów postawionych na zajęciach laboratoryjnych				
S-3	P	Test zaliczeniowy				
Zamierzone efekty kształcenia		Odniesienie do efektów kształcenia dla kierunku studiów	Cel przedmiotu	Treści programowe	Metody nauczania	Sposób oceny



Wiedza

PESWa_10_01_W01 Wie czym jest Linux i jakie ma zastosowanie, zna pliki i polecenia do konfiguracji systemu	PESWa_10_W01	C-1	T-W-1 T-W-2	M-1 M-2 M-4 M-5	S-1 S-2 S-3
PESWa_10_01_W02 Potrafi płynnie poruszać się po systemie i nim zarządzać	PESWa_10_W02	C-1	T-W-3 T-W-4	M-1 M-2	S-3

Umiejętności

PESWa_10_01_U01 Potrafi zainstalować system Linux i go spersonalizować	PESWa_10_U01	C-1	T-L-1 T-L-2	M-4 M-5	S-1 S-2
PESWa_10_01_U02 Umie automatyzować powtarzalne czynności w powłoce	PESWa_10_U01	C-2	T-L-3 T-L-4	M-1 M-2 M-3 M-4 M-5	S-1 S-2 S-3

Inne kompetencje społeczne i personalne

Efekt	Ocena	Kryterium oceny
-------	-------	-----------------

Wiedza

PESWa_10_01_W01	2,0	
	3,0	Wie czym jest Linux i jakie ma zastosowanie, zna pliki i podstawowe polecenia do konfiguracji systemu.
	3,5	
	4,0	
	4,5	
	5,0	
PESWa_10_01_W02	2,0	
	3,0	Wie jak poruszać się po systemie.
	3,5	
	4,0	
	4,5	
	5,0	

Umiejętności

PESWa_10_01_U01	2,0	
	3,0	Potrafi omówić znaczenie poszczególnych katalogów i system uprawnień w systemie Linux
	3,5	
	4,0	
	4,5	
	5,0	
PESWa_10_01_U02	2,0	
	3,0	Potrafi stworzyć proste skrypty w systemie Linux
	3,5	
	4,0	
	4,5	
	5,0	

Inne kompetencje społeczne i personalne

Literatura podstawowa

1. Paul Cobbaut, Linux Fundamentals,, Netsec BVBA, Paul Cobbaut, 2015, <http://linux-training.be/>

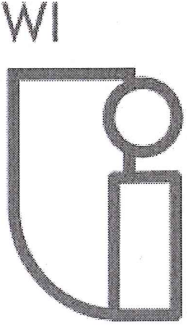
Literatura uzupełniająca

1. Richard Petersen, The Complete Reference Linux, Mc Graw Hill, 2015



Zachodniopomorski Uniwersytet Technologiczny w Szczecinie

Wydział Informatyki

Kierunek studiów	Programowanie z elementami systemów wbudowanych						
Forma studiów	niestacjonarna	Poziom	podyplomowy				
Obszary studiów	nauki techniczne						
Profil	-						
Moduł							
Przedmiot	Wstęp do programowania C w systemach wbudowanych						
Kod	/SP/PESW/02						
Jednostka prowadząca	Katedra Architektury Komputerów i Teleinformatyki						
ECTS	3,0	ECTS (formy)	3,0				
Forma zaliczenia	egzamin	Język	polski				
Forma dydaktyczna	Kod	Semestr	Godziny	ECTS	Waga	Zaliczenie	
wykłady	W	1	8	1,0	0,60	egzamin	
laboratoria	L	1	12	2,0	0,40	zaliczenie	
Nauczyciel odpowiedzialny	Makles Krzysztof (Krzysztof.Makles@zut.edu.pl)						
Inni nauczyciele	Maciaszczyk Radosław (Radoslaw.Maciaszczyk@zut.edu.pl)						
Wymagania wstępne							
W-1	Elementarna wiedza w zakresie programowania w jakimkolwiek języku programowania						
Cele modułu/przedmiotu							
C-1	Słuchacz zna tematykę kompilacji i kros-kompilacji oraz budowania i instalacji Linux na dedykowanej platformie						
Treści programowe z podziałem na formy zajęć						Liczba godzin	
T-W-1	Omówienie elementów języka C					4	
T-W-2	Architektura systemu Linux z punktu widzenia programisty					1	
T-W-3	Podstawowe narzędzia programistyczne: kompilator (kros-kompilator), linker, proces budowania, system kontroli wersji					1	
T-W-4	Implementacja i instalacji aplikacji na platformę dedykowaną					2	
T-L-1	Budowanie i instalacja Linux na maszynie docelowej					2	
T-L-2	Implementacja i instalacja podstawowej aplikacji na maszynie docelowej					6	
T-L-3	Implementacja i instalacja aplikacji i bibliotek dynamicznie dołączanych na maszynie docelowej					2	
T-L-4	Automatyzacja procesu budowania z wykorzystaniem GNU Make					2	
Obciążenie pracą studenta - formy aktywności						Liczba godzin	
A-W-1	Udział w wykładach					8	
A-W-2	Samodzielnie studiowanie zalecanej literatury i analiza materiału z wykładów					20	
A-W-3	Konsultacje					2	
A-L-1	Uczestnictwo w zajęciach					12	
A-L-2	Przygotowanie do zajęć					18	
A-L-3	Samodzielne rozwiązywanie zadań i problemów postawionych na laboratoriach					30	
Metody nauczania / narzędzia dydaktyczne							
M-1	Wykład problemowy						
M-2	Wykład informacyjny						
M-3	Anegdota						
M-4	Ćwiczenia laboratoryjne						
M-5	Nauka i ćwiczenia własne						
Sposoby oceny (F - formująca, P - podsumowująca)							
S-1	F	Obserwacja aktywności na zajęciach laboratoryjnych					
S-2	F	Ocena kompletności rozwiązań problemów postawionych na zajęciach laboratoryjnych					
S-3	P	Test zaliczeniowy					
Zamierzone efekty kształcenia			Odniesienie do efektów kształcenia dla kierunku studiów	Cel przedmiotu	Treści programowe	Metody nauczania	Sposób oceny



Wiedza

PESWa_10-02_W01 Zna architekturę systemu Linux, podstawowe narzędzia programistyczne	PESWa_10-W03	C-1		M-1 M-2 M-4 M-5	S-1 S-2 S-3
PESWa_10-02_W02 Wie jak implementować i wdrażać aplikacje na platformy dedykowane	PESWa_10-W04	C-1		M-1 M-2	S-3

Umiejętności

PESWa_10-02_U01 Umie implementować i instalować aplikacje na platformie dedykowanej	PESWa_10-U02	C-1		M-1 M-2 M-3 M-4 M-5	S-1 S-2 S-3
--	--------------	-----	--	---------------------------------	-------------------

Inne kompetencje społeczne i personalne

PESWa_10-02_K01 Rozumie rolę programisty w procesie wytwarzania oprogramowania	PESWa_10-K01	C-1		M-4 M-5	S-1
---	--------------	-----	--	------------	-----

Efekt	Ocena	Kryterium oceny			
-------	-------	-----------------	--	--	--

Wiedza

PESWa_10-02_W01	2,0	
	3,0	Zna podstawowe narzędzia systemu Linux
	3,5	
	4,0	
	4,5	
	5,0	
PESWa_10-02_W02	2,0	
	3,0	Potrafi wymienić elementy niezbędne do wdrożenia prostej aplikacji
	3,5	
	4,0	
	4,5	
	5,0	

Umiejętności

PESWa_10-02_U01	2,0	
	3,0	Potrafi wdrożyć prostą aplikację
	3,5	
	4,0	
	4,5	
	5,0	

Inne kompetencje społeczne i personalne

PESWa_10-02_K01	2,0	
	3,0	
	3,5	
	4,0	
	4,5	
	5,0	

Literatura podstawowa

1. Brian W. Kernighan, Dennis M. Ritchie, Język ANSI C, WNT, 2007

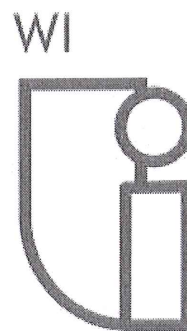
Literatura uzupełniająca

1. Robert Mecklenburg, Managing Projects with GNU Make, O'Reilly, 2004



Zachodniopomorski Uniwersytet Technologiczny w Szczecinie

Wydział Informatyki



Kierunek studiów		Programowanie z elementami systemów wbudowanych						
Forma studiów		niestacjonarna	Poziom	podyplomowy				
Obszary studiów		nauki techniczne						
Profil		-						
Moduł								
Przedmiot		Dynamiczne zarządzanie pamięcią, algorytmy i struktury danych						
Kod		/SP/PESW/03						
Jednostka prowadząca		Katedra Architektury Komputerów i Teleinformatyki						
ECTS		2,0	ECTS (formy)	2,0				
Forma zaliczenia		zaliczenie	Język	polski				
Forma dydaktyczna		Kod	Semestr	Godziny	ECTS	Waga	Zaliczenie	
wykłady		W	1	4	0,6	0,30	zaliczenie	
laboratoria		L	1	10	1,4	0,70	zaliczenie	
Nauczyciel odpowiedzialny		Maciaszczyk Radosław (Radoslaw.Maciaszczyk@zut.edu.pl)						
Inni nauczyciele								
Wymagania wstępne								
W-1	Słuchacz zna zasady konstruowania algorytmów							
Cele modułu/przedmiotu								
C-1	Student potrafi analizować i badać złożoność algorytmów, zna podstawowe algorytmy wykorzystywane w informatyce oraz potrafi je zaimplementować z użyciem dynamicznego zarządzania pamięcią. Zna różnicę między statyczną i dynamiczną alokacją pamięci.							
Treści programowe z podziałem na formy zajęć							Liczba godzin	
T-W-1	Wprowadzanie do algorytmów.						1	
T-W-2	Algorytmy sortowania, algorytmy wyszukiwania.						1	
T-W-3	Dynamiczne zarządzanie pamięcią.						1	
T-W-4	Struktury danych, algorytmy zaawansowane.						1	
T-L-1	Nauka notacji, zapisu algorytmów i badanie złożoności.						2	
T-L-2	Implementacja znanych algorytmów sortowania.						2	
T-L-3	Implementacja znanych algorytmów wyszukiwania.						2	
T-L-4	Wykorzystanie dynamicznych sposobów zarządzania pamięcią.						2	
T-L-5	Implementacja znanych struktur danych i algorytmów z nimi związanych.						2	
Obciążenie pracą studenta - formy aktywności							Liczba godzin	
A-W-1	Uczestnictwo w zajęciach						4	
A-W-2	Samodzielnie studiowanie zalecanej literatury i analiza materiału z wykładów						12	
A-W-3	Konsultacje						2	
A-L-1	Uczestnictwo w zajęciach						10	
A-L-2	Przygotowanie do zajęć						10	
A-L-3	Samodzielne rozwiązywania zadań i problemów postawionych na laboratoriach.						22	
Metody nauczania / narzędzia dydaktyczne								
M-1	Wykład problemowy							
M-2	Wykład informacyjny							
M-3	Anegdota							
M-4	Ćwiczenia laboratoryjne							
M-5	Nauka i ćwiczenia własne							
Sposoby oceny (F - formująca, P - podsumowująca)								
S-1	F	Obserwacja aktywności na zajęciach laboratoryjnych						
S-2	F	Ocena kompletności rozwiązań problemów postawionych na zajęciach laboratoryjnych						
S-3	F	Test zaliczeniowy						



Wydział Informatyki

Zamierzone efekty kształcenia	Odniesienie do efektów kształcenia dla kierunku studiów	Cel przedmiotu	Treści programowe	Metody nauczania	Sposób oceny
Wiedza					
PESWa_10-_03_W01 Zna podstawowe mechanizmy dynamicznego zarządzania pamięcią.	PESWa_10-_W05	C-1	T-L-3 T-W-3	M-1 M-2 M-3 M-4	S-2 S-3
PESWa_10-_03_W02 Zna podstawowe struktury danych i podstawowe algorytmy stosowane w informatyce.	PESWa_10-_W06	C-1	T-L-1 T-W-1 T-L-2 T-W-2 T-L-3 T-W-4 T-L-5	M-1 M-2 M-3 M-4 M-5	S-1 S-2 S-3
Umiejętności					
PESWa_10-_03_U01 Potrafi wykorzystać mechanizmy dynamicznego zarządzania pamięcią.	PESWa_10-_U03	C-1	T-L-4 T-W-4	M-1 M-2 M-4 M-5	S-1 S-2 S-3
PESWa_10-_03_U02 Potrafi zaimplementować podstawowe algorytmy i struktury danych stosowane w informatyce	PESWa_10-_U04	C-1	T-L-1 T-L-5 T-L-2 T-W-1 T-L-3 T-W-4		S-1 S-2 S-3
Inne kompetencje społeczne i personalne					

Efekt	Ocena	Kryterium oceny
Wiedza		
PESWa_10-_03_W01	2,0	
	3,0	Zna metody wykorzystywania dynamicznego zarządzania pamięcią.
	3,5	
	4,0	
	4,5	
	5,0	
PESWa_10-_03_W02	2,0	
	3,0	Zna metody wykorzystywania struktur danych, Zna metody zapisu algorytmów.
	3,5	
	4,0	Rozumie różnice między strukturami danych oraz różnicę pomiędzy algorytmami sortowania, a wyszukiwania.
	4,5	
	5,0	Potrafi wybrać optymalną strukturę danych i algorytm dla zadanych problemów.
Umiejętności		
PESWa_10-_03_U01	2,0	
	3,0	Potrafi wykorzystać metody wykorzystywania dynamicznego zarządzania pamięcią.
	3,5	
	4,0	Potrafi wykorzystać różnice między statycznym i dynamicznym zarządzaniem pamięcią.
	4,5	
	5,0	Potrafi zaimplementować optymalny sposób zarządzania pamięcią dla zadanych problemów.
PESWa_10-_03_U02	2,0	
	3,0	Potrafi wykorzystać metody zapisu algorytmów i użyć odpowiednie struktury danych.
	3,5	
	4,0	Potrafi wykorzystać różnice między strukturami danych i użyć metod sortowania lub wyszukiwania
	4,5	
	5,0	Potrafi zaimplementować optymalną strukturę danych dla zadanych problemów.

Inne kompetencje społeczne i personalne

Literatura podstawowa

1. Harel David, Feldman Yishai, Rzecz o istocie informatyki. Algorytmika. Klasyka informatyki;

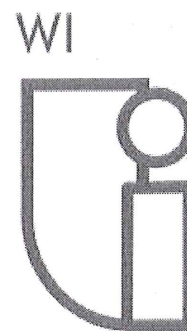
Literatura uzupełniająca

1. Prof. Erik Demaine, Introduction to Algorithms



Zachodniopomorski Uniwersytet Technologiczny w Szczecinie

Wydział Informatyki



Kierunek studiów		Programowanie z elementami systemów wbudowanych				
Forma studiów		niestacjonarna	Poziom	podyplomowy		
Obszary studiów		nauki techniczne				
Profil		-				
Moduł						
Przedmiot		Wstęp do programowania systemowego w Linux				
Kod		/SP/PESW/04				
Jednostka prowadząca		Katedra Architektury Komputerów i Teleinformatyki				
ECTS		3,0	ECTS (formy)	3,0		
Forma zaliczenia		egzamin	Język	polski		
Forma dydaktyczna	Kod	Semestr	Godziny	ECTS	Waga	Zaliczenie
wykłady	W	1	6	1,0	0,40	egzamin
laboratoria	L	1	12	2,0	0,60	zaliczenie
Nauczyciel odpowiedzialny		Maciaszczyk Radosław (Radoslaw.Maciaszczyk@zut.edu.pl)				
Inni nauczyciele						
Wymagania wstępne						
W-1	Słuchacz zna składnię języka C					
Cele modułu/przedmiotu						
C-1	Słuchacz zna mechanizmy systemowe Linux na rzecz implementacji złożonych aplikacji					
Treści programowe z podziałem na formy zajęć						Liczba godzin
T-W-1	Podstawowa nomenklatura: aplikacja, program, proces, wątek. Idea wieloprocusowości.					1
T-W-2	Procesy w systemie Linux.					1
T-W-3	Wielowątkowość i synchronizacja.					2
T-W-4	Komunikacja międzyprocesowa i międzymaszynowa.					2
T-L-1	Tworzenie procesów z wykorzystaniem fork					2
T-L-2	Fork i exec* - zarządzanie procesami					2
T-L-3	Tworzenie i zamykanie wątków					2
T-L-4	Synchronizacja międzywątkowa					2
T-L-5	Komunikacja międzyprocesowa w oparciu o gniazda					2
T-L-6	Komunikacja międzyprocesowa i demultipleksacja zdarzeń					2
Obciążenie pracą studenta - formy aktywności						Liczba godzin
A-W-1	Uczestnictwo w wykładach					6
A-W-2	Samodzielnie studiowanie zalecanej literatury i analiza materiału z wykładów					22
A-W-3	Konsultacje					2
A-L-1	Uczestnictwo w zajęciach					12
A-L-2	Przygotowanie do zajęć					18
A-L-3	Samodzielne rozwiązywania zadań i problemów postawionych na laboratoriach.					30
Metody nauczania / narzędzia dydaktyczne						
M-1	Wykład problemowy					
M-2	Wykład informacyjny					
M-3	Anegdota					
M-4	Ćwiczenia laboratoryjne					
M-5	Nauka i ćwiczenia własne					
Sposoby oceny (F - formująca, P - podsumowująca)						
S-1	F	Obserwacja aktywności na zajęciach laboratoryjnych				
S-2	F	Ocena kompletności rozwiązań problemów postawionych na zajęciach laboratoryjnych				
S-3	P	Test zaliczeniowy				



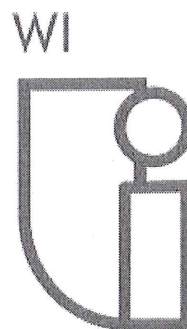
Wydział Informatyki

Zamierzone efekty kształcenia	Odniesienie do efektów kształcenia dla kierunku studiów	Cel przedmiotu	Treści programowe	Metody nauczania	Sposób oceny
Wiedza					
PESWa_10_04_W01 Wie jakie mechanizmy Linux udostępnia programiście oraz wie jak je zaimplementować w aplikacji	PESWa_10_W04	C-1	T-W-1 T-W-3 T-W-2 T-W-4	M-1 M-2 M-3	S-1 S-2 S-3
Umiejętności					
PESWa_10_04_U01 Umie stosować odpowiednie mechanizmy systemu Linux w zależności od potrzeb aplikacji	PESWa_10_U12	C-1	T-L-1 T-L-4 T-L-2 T-L-5 T-L-3 T-L-6	M-3 M-4 M-5	S-1 S-2
Inne kompetencje społeczne i personalne					
Efekt	Ocena	Kryterium oceny			
Wiedza					
PESWa_10_04_W01	2,0				
	3,0	Zna podstawową nomenklaturę stosowaną przy programowaniu w systemie Linux			
	3,5				
	4,0				
	4,5				
	5,0				
Umiejętności					
PESWa_10_04_U01	2,0				
	3,0	Potrafi stworzyć proste wielowatkowe programy.			
	3,5				
	4,0				
	4,5				
	5,0				
Inne kompetencje społeczne i personalne					
Literatura podstawowa					
1. CodeSourcery LLC, 1. Advanced Linux Programming					
Literatura uzupełniająca					
1. Sielberschatz, Galvin, Gagne, Operating System Concepts					



Zachodniopomorski Uniwersytet Technologiczny w Szczecinie

Wydział Informatyki



Kierunek studiów		Programowanie z elementami systemów wbudowanych					
Forma studiów		niestacjonarna	Poziom	podyplomowy			
Obszary studiów		nauki techniczne					
Profil		-					
Moduł							
Przedmiot		Komunikacja międzyprocesowa i międzywątkowa					
Kod		/SP/PESW/05					
Jednostka prowadząca		Katedra Architektury Komputerów i Teleinformatyki					
ECTS		2,0	ECTS (formy)	2,0			
Forma zaliczenia		zaliczenie	Język	polski			
Forma dydaktyczna	Kod	Semestr	Godziny	ECTS	Waga	Zaliczenie	
wykłady	W	1	4	0,6	0,40	zaliczenie	
laboratoria	L	1	8	1,4	0,60	zaliczenie	
Nauczyciel odpowiedzialny		Maciaszczyk Radosław (Radoslaw.Maciaszczyk@zut.edu.pl)					
Inni nauczyciele							
Wymagania wstępne							
W-1	Słuchacz zna zasady programowania w C/C++						
W-2	Słuchacz zna podstawy systemów operacyjnych (wątki, procesy) oraz podstawy obsługi systemu Linux.						
Cele modułu/przedmiotu							
C-1	Słuchacz zna metody komunikacji między wątkami i procesami w systemie Linux, rozumie różnice między nimi oraz jest w stanie dobrać odpowiednią metodę do zadanego problemu.						
Treści programowe z podziałem na formy zajęć						Liczba godzin	
T-W-1	Wstęp do komunikacji międzyprocesowej, tworzenie wątków i procesów.					2	
T-W-2	Podstawowe problemy w komunikacji międzyprocesowej: zakleszczenia, zagłodzenie wątków.					2	
T-L-1	Wstęp do komunikacji międzyprocesowej, tworzenie wątków i procesów.					1	
T-L-2	Podstawowe problemy w komunikacji międzyprocesowej: zakleszczenia, zagłodzenie wątków.					1	
T-L-3	Mechanizmy komunikacji międzyprocesowej i synchronizacji: - mutexy - semafony - zmienne warunkowe - kolejki - potoki - pamięć współdzielona					4	
T-L-4	Wprowadzenie do mechanizmów komunikacji w bibliotece standardowej					2	
Obciążenie pracą studenta - formy aktywności						Liczba godzin	
A-W-1	Udział w wykładach					4	
A-W-2	Samodzielnie studiowanie zalecanej literatury i analiza materiału z wykładów					14	
A-L-1	Uczestnictwo w zajęciach					8	
A-L-2	Przygotowanie do zajęć					12	
A-L-3	Samodzielne rozwiązywanie zadań i problemów postawionych przez prowadzącego.					20	
A-L-4	Konsultacje					2	
Metody nauczania / narzędzia dydaktyczne							
M-1	Wykład problemowy						
M-2	Wykład informacyjny						
M-3	Anegdota						
M-4	Ćwiczenia						
M-5	Nauka i ćwiczenia własne						
Sposoby oceny (F - formująca, P - podsumowująca)							
S-1	P	Obserwacja aktywności na zajęciach					
S-2	P	Ocena kompletności rozwiązań problemów postawionych na zajęciach i wykonanych samodzielnie w domu					
S-3	F	Test zaliczeniowy					
Zamierzone efekty kształcenia			Odniesienie do efektów kształcenia dla kierunku studiów	Cel przedmiotu	Treści programowe	Metody nauczania	Sposób oceny
Wiedza							



Wydział Informatyki

PESWa_10_05_W01 Zna metody tworzenia procesów i wątków lub sposoby komunikacji pomiędzy nimi w systemie Linux.	PESWa_10_W05	C-1	T-L-1 T-L-2 T-L-3	T-L-4 T-W-1 T-W-2	M-1 M-2 M-3 M-4 M-5	S-1 S-2 S-3
---	--------------	-----	-------------------------	-------------------------	---------------------------------	-------------------

Umiejętności

PESWa_10_05_U01 Umie zaimplementować komunikację między procesami i wątkami w systemie Linux.	PESWa_10_U05	C-1	T-L-3 T-L-4	T-W-2	M-2 M-3 M-4 M-5	S-1 S-2 S-3
--	--------------	-----	----------------	-------	--------------------------	-------------------

Inne kompetencje społeczne i personalne

Efekt	Ocena	Kryterium oceny
-------	-------	-----------------

Wiedza

PESWa_10_05_W01	2,0	
	3,0	Zna metody tworzenia wątków w systemie Linux.
	3,5	
	4,0	Zna metody tworzenia procesów i wątków i komunikacji w systemie Linux.
	4,5	
	5,0	Rozumie różnicę pomiędzy wątkami i procesami w systemie Linux oraz potrafi dobrać optymalne rozwiązanie do postawionego problemu. Zna metody komunikacji dostępne w bibliotece standardowej.

Umiejętności

PESWa_10_05_U01	2,0	
	3,0	Potrafi zaimplementować przynajmniej w jeden sposób komunikację pomiędzy wątkami i procesami w systemie Linux.
	3,5	
	4,0	Potrafi zaimplementować wszystkie omawiane metody komunikacji pomiędzy wątkami i procesami w systemie Linux.
	4,5	
	5,0	Potrafi zaimplementować komunikację pomiędzy procesami przy użyciu biblioteki standardowej.

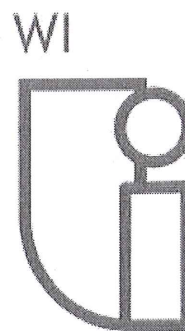
Inne kompetencje społeczne i personalne

Literatura podstawowa

1. Anthony Williams, Język C++ i przetwarzanie współbieżne w akcji, Helion, 2013



Wydział Informatyki



Kierunek studiów	Programowanie z elementami systemów wbudowanych					
Forma studiów	niestacjonarna	Poziom	podyplomowy			
Obszary studiów	nauki techniczne					
Profil	-					
Moduł						
Przedmiot	Programowanie obiektowe - wstęp do C++					
Kod	/SP/PESW/06					
Jednostka prowadząca	Katedra Architektury Komputerów i Teleinformatyki					
ECTS	5,0	ECTS (formy)	5,0			
Forma zaliczenia	zaliczenie	Język	polski			
Forma dydaktyczna	Kod	Semestr	Godziny	ECTS	Waga	Zaliczenie
wykłady	W	1	12	1,8	0,35	zaliczenie
laboratoria	L	1	20	3,2	0,65	zaliczenie
Nauczyciel odpowiedzialny	Mościcki Mirosław (Miroslaw.Moscicki@zut.edu.pl)					
Inni nauczyciele	Maciaszczyk Radosław (Radoslaw.Maciaszczyk@zut.edu.pl)					
Wymagania wstępne						
W-1	Słuchacz zna podstawową składnię języka C. Słuchacz potrafi czytać pseudokod w podstawowym zakresie					
Cele modułu/przedmiotu						
C-1	Słuchacz zna leksykę i składnię języka C++ w zakresie niezbędnym do samodzielnego konstruowania prostych programów w imperatywie proceduralnym, rozumie proces tworzenia oprogramowania (kompilacja), potrafi opisać sposób uruchomienia programów przez komputer.					
Treści programowe z podziałem na formy zajęć						Liczba godzin
T-W-1	Wyjaśnienie czym jest programowanie, jak działa komputer, na czym polega kompilacja oraz linkowanie. Objasnienie sposobu uruchamiania programów przez system operacyjny.					2
T-W-2	Proces kompilacji, objaśnienie podstawowych elementów kompilatora; Wstęp do debuggera					1
T-W-3	Podstawy języka C++: funkcja main, typy wbudowane, zmienne, napisy, funkcje, enumeracje, tablice, instrukcje sterujące, wejście, wyjście					3
T-W-4	Programowanie funkcyjne w C++: funkcje, operatory, przeladowowania funkcji, zakres życia zmiennych, przestrzenie nazw					3
T-W-5	Obsługa błędów w C++: wyjątki, kody błędów					1
T-W-6	Typy własne w C++: klasy, konstruktory, destruktory, listy inicjalizacyjne, metody, składowe, zakres widoczności					2
T-L-1	Zapoznanie z wierszem poleceń, kompilatorem					1
T-L-2	Implementacja prostych programów implementowanych w stylu proceduralnym					6
T-L-3	Implementacja prostych programów z wykorzystaniem wejścia i wyjścia					6
T-L-4	Implementacja złożonych programów implementowanych w stylu proceduralnym z obsługą błędów					6
T-L-5	Techniki debuggowania programów					1
Obciążenie pracą studenta - formy aktywności						Liczba godzin
A-W-1	Udział w wykładach					12
A-W-2	Samodzielnie studiowanie zalecanej literatury i analiza materiału z wykładów					40
A-W-3	Konsultacje					2
A-L-1	Uczestnictwo w zajęciach					20
A-L-2	Przygotowanie do zajęć					26
A-L-3	Samodzielne rozwiązywania zadań i problemów postawionych na laboratoriach					50
Metody nauczania / narzędzia dydaktyczne						
M-1	Wykład problemowy					
M-2	Wykład informacyjny					
M-3	Anegdota					
M-4	Ćwiczenia laboratoryjne					
M-5	Nauka i ćwiczenia własne					
Sposoby oceny (F - formująca, P - podsumowująca)						
S-1	F	Obserwacja aktywności na zajęciach laboratoryjnych				



Wydział Informatyki

Sposoby oceny (F - formująca, P - podsumowująca)

S-2	F	Ocena kompletności rozwiązań problemów postawionych na zajęciach laboratoryjnych
S-3	P	Test zaliczeniowy

Zamierzone efekty kształcenia	Odniesienie do efektów kształcenia dla kierunku studiów	Cel przedmiotu	Treści programowe	Metody nauczania	Sposób oceny
Wiedza					
PESWa_10_06_W01 Zna podstawowe techniki programowania proceduralnego w języku C++, zna składnię, semantykę i sferę zastosowań języka C++.	PESWa_10_W09	C-1	T-L-1 T-W-4 T-L-2 T-W-5 T-L-3 T-W-6 T-W-3	M-1 M-2 M-4 M-5	S-1 S-2 S-3
PESWa_10_06_W02 Rozumie proces kompilacji oraz uruchamiania programów	PESWa_10_W10	C-1	T-W-1 T-W-2	M-1 M-2	S-3
Umiejętności					
PESWa_10_06_U01 Umie sterować procesem kompilacji	PESWa_10_U06	C-1	T-L-4 T-W-2	M-1 M-2 M-3 M-4 M-5	S-1 S-2 S-3
PESWa_10_06_U02 Umie pisać i uruchamiać programy w języku C++	PESWa_10_U07	C-1	T-L-2 T-W-3 T-L-3 T-W-4 T-L-4 T-W-5 T-W-2 T-W-6	M-1 M-2 M-3 M-4 M-5	S-1 S-2 S-3
Inne kompetencje społeczne i personalne					

Efekt	Ocena	Kryterium oceny
Wiedza		
PESWa_10_06_W01	2,0	
	3,0	Analizuje przedstawiony kod źródłowy w języku C++ i potrafi wskazać efekty jego działania. Zna składnię i semantykę języka C++, umie analizować przedstawiony mu kod.
	3,5	
	4,0	Poprawnie implementuje w języku C++ proste algorytmy, Potrafi rozpoznać potencjalne problemy w przedstawionym rozwiązaniu problemu
	4,5	
	5,0	Poprawnie implementuje w języku C++ złożone algorytmy; potrafi wyjaśnić różnice w wydajności pomiędzy różnymi możliwymi implementacjami zadanych problemów
PESWa_10_06_W02	2,0	
	3,0	Rozumie proces kompilacji programów złożonych z jednego pliku źródłowego
	3,5	
	4,0	Rozumie proces kompilacji programów złożonych z wielu plików źródłowych
	4,5	
	5,0	
Umiejętności		
PESWa_10_06_U01	2,0	
	3,0	Potrafi skompilować program składający się z jednego pliku źródłowego
	3,5	
	4,0	Potrafi naprawiać błędy składniowe oraz proste błędy semantyczne. Potrafi skompilować program składający się z wielu plików źródłowych
	4,5	
	5,0	Rozumie proces kompilacji, potrafi poprawnie dobrać flagi kompilacji, umie posługiwać się debuggerem
PESWa_10_06_U02	2,0	
	3,0	Potrafi posługiwać się wybranym środowiskiem zintegrowanym w celu implementacji i uruchomienia własnego kodu
	3,5	
	4,0	
	4,5	
	5,0	
Inne kompetencje społeczne i personalne		

Literatura podstawowa

1. Bjarne Stroustrup, The C++ Programming Language, Addison-Wesley, 2013

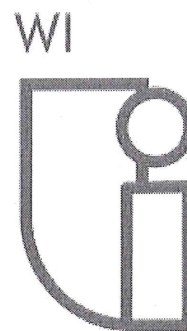
Literatura uzupełniająca

1. Scott Meyers, Effective Modern C++, O'Reilly Media, 2014



Zachodniopomorski Uniwersytet Technologiczny w Szczecinie

Wydział Informatyki



Kierunek studiów		Programowanie z elementami systemów wbudowanych				
Forma studiów		niestacjonarna	Poziom	podyplomowy		
Obszary studiów		nauki techniczne				
Profil		-				
Moduł						
Przedmiot		Zaawansowane programowanie obiektowe C++				
Kod		/SP/PESW/07				
Jednostka prowadząca		Katedra Architektury Komputerów i Teleinformatyki				
ECTS		4,5	ECTS (formy)	4,5		
Forma zaliczenia		egzamin	Język	polski		
Forma dydaktyczna	Kod	Semestr	Godziny	ECTS	Waga	Zaliczenie
wykłady	W	2	12	2,0	0,40	zaliczenie
laboratoria	L	2	16	2,5	0,60	egzamin
Nauczyciel odpowiedzialny		Maciaszczyk Radosław (Radoslaw.Maciaszczyk@zut.edu.pl)				
Inni nauczyciele						
Wymagania wstępne						
W-1	Słuchacz zna składnię języka C++, potrafi implementować proste algorytmy w języku C++, potrafi kompilować programy składające się z wielu plików źródłowych					
Cele modułu/przedmiotu						
C-1	Słuchacz potrafi implementować algorytmy z wykorzystaniem paradygmatu programowania obiektowego w języku C++. Słuchacz potrafi pisać programy równoległe					
Treści programowe z podziałem na formy zajęć						Liczba godzin
T-W-1	Powtórzenie wiadomości o typach własnych: tworzenie, kopiowanie, przesuwanie oraz niszczenie. Powtórzenie wiadomości o czasie życia zmiennych					1
T-W-2	Przeładowywanie operatorów dla typów własnych oraz specjalne operatory.					2
T-W-3	Dziedziczenie oraz hierarchie klas. Polimorfizm dynamiczny w kontekście języka C++. Omówienie mechanizmu RTTI					3
T-W-4	Enkapsulacja danych w języku C++, widoczność pól klasy w drzewie dziedziczenia					2
T-W-5	Wprowadzenie do programowania generycznego dla języka C++. Klasy szablonowe, funkcje szablonowe, specjalizacja, częściowa specjalizacja, przeciążenia.					2
T-W-6	Programowanie wielowątkowe w języku C++ z wykorzystaniem mechanizmów z przestrzeni nazw std					2
T-L-1	Implementacja prostych algorytmów z wykorzystaniem paradygmatu programowania obiektowego					4
T-L-2	Implementacja algorytmów wykorzystujących dziedziczenie, polimorfizm					4
T-L-3	Implementacja prostych algorytmów wykorzystujących paradygmat programowania generycznego					4
T-L-4	Implementacja algorytmów wykorzystujących wielowątkowość					4
Obciążenie pracą studenta - formy aktywności						Liczba godzin
A-W-1	Udział w wykładach					12
A-W-2	Samodzielnie studiowanie zalecanej literatury i analiza materiału z wykładów					46
A-W-3	Konsultacje					2
A-L-1	Uczestnictwo w zajęciach					16
A-L-2	Przygotowanie do zajęć					24
A-L-3	Samodzielne rozwiązywania zadań i problemów postawionych na laboratoriach					36
Metody nauczania / narzędzia dydaktyczne						
M-1	Wykład problemowy					
M-2	Wykład informacyjny					
M-3	Anegdota					
M-4	Ćwiczenia laboratoryjne					
M-5	Nauka i ćwiczenia własne					
Sposoby oceny (F - formująca, P - podsumowująca)						
S-1	F	Obserwacja aktywności na zajęciach laboratoryjnych				
S-2	F	Ocena kompletności rozwiązań problemów postawionych na zajęciach laboratoryjnych				



Wydział Informatyki

Sposoby oceny (F - formująca, P - podsumowująca)

S-3	P	Test zaliczeniowy
-----	---	-------------------

Zamierzone efekty kształcenia	Odniesienie do efektów kształcenia dla kierunku studiów	Cel przedmiotu	Treści programowe	Metody nauczania	Sposób oceny
-------------------------------	---	----------------	-------------------	------------------	--------------

Wiedza

PESWa_10-07_W01 Zna paradygmaty programowania obiektowego i generycznego w kontekście języka C++	PESWa_10-W11	C-1	T-L-1 T-W-4 T-L-2 T-W-5 T-L-3 T-W-6 T-W-3	M-1 M-2 M-3 M-4 M-5	S-1 S-2 S-3
---	--------------	-----	--	---------------------------------	-------------------

Umiejętności

PESWa_10-07_U01 Umie rozwiązać algorytm wykorzystując implementację korzystającą z paradygmatu programowania obiektowego lub generycznego	PESWa_10-U07	C-1	T-L-4 T-W-4 T-W-2 T-W-5 T-W-3 T-W-6	M-1 M-2 M-3 M-4 M-5	S-1 S-2 S-3
PESWa_10-07_U02 Umie zrównoleglic zadany algorytm wykorzystując mechanizm programowania wielowątkowego w języku C++	PESWa_10-U08	C-1	T-W-2 T-W-5 T-W-3 T-W-6 T-W-4	M-1 M-2 M-3	S-3

Inne kompetencje społeczne i personalne

Efekt	Ocena	Kryterium oceny
-------	-------	-----------------

Wiedza

PESWa_10-07_W01	2,0	
	3,0	Potrafi wskazać główne właściwości programowania obiektowego i generycznego w języku C++
	3,5	
	4,0	Potrafi wyjaśnić polimorfizm w kontekście języka C++, oraz techniki związane z programowaniem generycznym
	4,5	
	5,0	Potrafi rozwiązać przedstawione problemy programowania wielowątkowego oraz wyjaśnić ich przyczynę

Umiejętności

PESWa_10-07_U01	2,0	
	3,0	Potrafi zaimplementować wybrany algorytm wykorzystując programowanie obiektowe
	3,5	
	4,0	
	4,5	
	5,0	
PESWa_10-07_U02	2,0	
	3,0	Potrafi użyć mechanizmów programowania wielowątkowego z języka C++
	3,5	
	4,0	
	4,5	
	5,0	

Inne kompetencje społeczne i personalne

Literatura podstawowa

1. Bjarne Stroustrup, The C++ Programming Language, Addison-Wesley, 2013

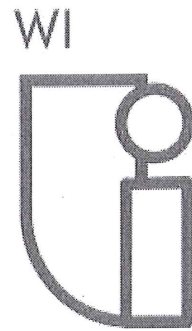
Literatura uzupełniająca

1. Scott Meyers, Effective Modern C++, O'Reilly Media, 2014



Zachodniopomorski Uniwersytet Technologiczny w Szczecinie

Wydział Informatyki



Kierunek studiów	Programowanie z elementami systemów wbudowanych					
Forma studiów	niestacjonarna	Poziom	podyplomowy			
Obszary studiów	nauki techniczne					
Profil	-					
Moduł						
Przedmiot	Biblioteka standardowa C++					
Kod	/SP/PESW/08					
Jednostka prowadząca	Katedra Architektury Komputerów i Teleinformatyki					
ECTS	4,0	ECTS (formy)	4,0			
Forma zaliczenia	zaliczenie	Język	polski			
Forma dydaktyczna	Kod	Semestr	Godziny	ECTS	Waga	Zaliczenie
wykłady	W	2	8	1,3	0,35	zaliczenie
laboratoria	L	2	16	2,7	0,65	zaliczenie
Nauczyciel odpowiedzialny	Maciaszczyk Radosław (Radoslaw.Maciaszczyk@zut.edu.pl)					
Inni nauczyciele						
Wymagania wstępne						
W-1	Słuchacz zna składnię języka C++, rozumienie mechanizm szablonów					
Cele modułu/przedmiotu						
C-1	Słuchacz zna interfejs kontenerów, algorytmów oraz iteratorów zawartych w bibliotece standardowej C++ oraz szczegóły implementacyjne pozwalające wybrać optymalny kontener bądź algorytm do rozwiązania powierzonego mu zadania					
Treści programowe z podziałem na formy zajęć						Liczba godzin
T-W-1	Wprowadzenie do teorii złożoności obliczeniowej					1
T-W-2	Objaśnienie interfejsu kontenerów ze szczegółami implementacyjnymi					3
T-W-3	Algorytmy sortujące, niemodyfikujące, modyfikujące					2
T-W-4	Wyjaśnienie idei iteratorów, podział iteratorów					1
T-W-5	Profilowanie kodu pod względem wyboru odpowiedniego algorytmu, kontenera					1
T-L-1	Zapoznanie ze środowiskiem, uruchamianie i analiza złożoności obliczeniowej gotowych prostych algorytmów na kontenerach					2
T-L-2	Implementacja prostych zadań z wykorzystaniem polimorfizmu oraz kontenerów					4
T-L-3	Implementacja prostych zadań obliczeniowych w oparciu o wbudowane algorytmy biblioteki standardowej					6
T-L-4	Analiza iteratorów, dedukcja typu iteratorów					1
T-L-5	Badanie wydajności kontenerów i algorytmów					3
Obciążenie pracą studenta - formy aktywności						Liczba godzin
A-W-1	Udział w wykładach					8
A-W-2	Konsultacje					2
A-W-3	Samodzielnie studiowanie zalecanej literatury i analiza materiału z wykładów					29
A-L-1	Uczestnictwo w zajęciach					16
A-L-2	Przygotowanie do zajęć					24
A-L-3	Samodzielne rozwiązywania zadań i problemów postawionych na laboratoriach					41
Metody nauczania / narzędzia dydaktyczne						
M-1	Wykład problemowy					
M-2	Wykład informacyjny					
M-3	Anegdota					
M-4	Ćwiczenia laboratoryjne					
M-5	Nauka i ćwiczenia własne					
Sposoby oceny (F - formująca, P - podsumowująca)						
S-1	F	Obserwacja aktywności na zajęciach laboratoryjnych				
S-2	F	Ocena kompletności rozwiązań problemów postawionych na zajęciach laboratoryjnych				
S-3	P	Test zaliczeniowy				



Wydział Informatyki

Zamierzone efekty kształcenia	Odniesienie do efektów kształcenia dla kierunku studiów	Cel przedmiotu	Treści programowe	Metody nauczania	Sposób oceny
Wiedza					
PESWa_10_08_W01 Zna podstawowe kontenery i algorytmy biblioteki standardowej C++	PESWa_10_W13	C-1	T-W-1 T-W-4 T-W-2 T-W-5 T-W-3	M-1 M-2 M-3	S-3
PESWa_10_08_W02 Ma wiedzę w zakresie interfejsu, implementacji i sfer zastosowań biblioteki standardowej	PESWa_10_W13	C-1	T-L-1 T-L-4 T-L-2 T-L-5 T-L-3	M-3 M-4 M-5	S-1 S-2 S-3
Umiejętności					
PESWa_10_08_U01 Potrafi wybrać odpowiedni kontener do przechowania elementów tego samego typu, jest w stanie mapować elementy ze sobą	PESWa_10_U09	C-1	T-L-4 T-W-4	M-1 M-2 M-3 M-4 M-5	S-1 S-2 S-3
PESWa_10_08_U02 Potrafi wybrać odpowiedni algorytm oraz kontener ze względu na wydajność.	PESWa_10_U09	C-1	T-W-2 T-W-4 T-W-3 T-W-5	M-1 M-2 M-3 M-5	S-3
Inne kompetencje społeczne i personalne					

Efekt	Ocena	Kryterium oceny
Wiedza		
PESWa_10_08_W01	2,0	
	3,0	Poprawnie klasyfikuje algorytmy i kontenery
	3,5	
	4,0	
	4,5	
	5,0	
PESWa_10_08_W02	2,0	
	3,0	Zna składnię i semantykę języka Python, umie analizować przedstawiony mu kod
	3,5	
	4,0	Zna szczegóły implementacyjne kontenerów
	4,5	
	5,0	Wie w jakich sytuacjach dobrać odpowiedni kontener czy algorytm
Umiejętności		
PESWa_10_08_U01	2,0	
	3,0	Umie dobrać odpowiedni kontener do przechowywania elementów tego samego typu
	3,5	
	4,0	Zna kontenery mapujące oraz umie mapować elementy ze sobą
	4,5	
	5,0	
PESWa_10_08_U02	2,0	
	3,0	Potrafi wybrać odpowiedni algorytm oraz kontener ze względu na wydajność
	3,5	
	4,0	
	4,5	
	5,0	
Inne kompetencje społeczne i personalne		

Literatura podstawowa

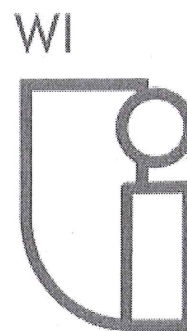
1. Nicolai M. Josuttis, C++. Biblioteka standardowa. Podręcznik programisty, Heilon

Literatura uzupełniająca

1. Scott Meyers, Effective STL



Wydział Informatyki



Kierunek studiów	Programowanie z elementami systemów wbudowanych					
Forma studiów	niestacjonarna	Poziom	podyplomowy			
Obszary studiów	nauki techniczne					
Profil	-					
Moduł						
Przedmiot	Testowanie oprogramowania - testy jednostowe/funkcjonalne					
Kod	/SP/PESW/09					
Jednostka prowadząca	Katedra Architektury Komputerów i Teleinformatyki					
ECTS	4,0	ECTS (formy)	4,0			
Forma zaliczenia	egzamin	Język	polski			
Forma dydaktyczna	Kod	Semestr	Godziny	ECTS	Waga	Zaliczenie
wykłady	W	2	8	1,3	0,35	egzamin
laboratoria	L	2	16	2,7	0,65	zaliczenie
Nauczyciel odpowiedzialny	Maciaszczyk Radosław (Radoslaw.Maciaszczyk@zut.edu.pl)					
Inni nauczyciele						
Wymagania wstępne						
W-1	Słuchacz zna podstawy programowania w języku C++ i języku skryptowym Python					
Cele modułu/przedmiotu						
C-1	Słuchacz zna podstawową terminologię i zagadnienia z dziedziny testowania programowania, umie posługiwać się narzędziami wykorzystywanymi w procesie wytwarzania oprogramowania, rozumie genezę powstawania błędów w oprogramowaniu i potrafi samodzielnie znaleźć i zgłosić powstałe błędy i tworzyć przypadki testowe zgodnie z wymaganiami.					
Treści programowe z podziałem na formy zajęć						Liczba godzin
T-W-1	Jakość w procesie wytwarzania oprogramowania					1
T-W-2	Omówienie definicji testowania oprogramowania. Skąd biorą się błędy. Dlaczego testowanie jest niezbędne. Podstawowe zasady testowania					1
T-W-3	Rodzaje testów. Poziomy testów. Statyczne techniki testowania					1
T-W-4	Identyfikacja warunków testowych. Organizacja i planowanie testów. Projektowanie i tworzenie przypadków testowych. Ustalenie formy raportowania wyników testów					1
T-W-5	Integracja ciągła: podstawowe informacje i środowisko					1
T-W-6	Techniki tworzenia przypadków testowych: techniki testowania, techniki czarnoskrzynkowe, techniki białoskrzynkowe, techniki oparte o incydent, techniki oparte o doświadczenie					2
T-W-7	Przykładowe techniki tworzenia przypadków testowych do analizy					1
T-L-1	Zapoznanie ze środowiskiem wersjonowania plików, uruchamianie i analiza prostych przykładów					2
T-L-2	Zapoznanie ze środowiskiem, uruchamianie i omówienie podstawowych zasad tworzenia przypadków testowych z wykorzystania narzędzi do unittestów					2
T-L-3	Wykorzystanie narzędzia do unittestów do tworzenia przykładowych przypadków testowych					4
T-L-4	Zapoznanie ze środowiskiem, uruchamianie i tworzenie przypadków testowych z wykorzystaniem narzędzia Selenium oraz Robotframework					8
Obciążenie pracą studenta - formy aktywności						Liczba godzin
A-W-1	Udział w wykładach					8
A-W-2	Konsultacje					2
A-W-3	Samodzielnie studiowanie zalecanej literatury i analiza materiału z wykładów					30
A-L-1	Uczestnictwo w zajęciach					16
A-L-2	Przygotowanie do zajęć					22
A-L-3	Samodzielne rozwiązywanie zadań i problemów postawionych na laboratoriach					44
Metody nauczania / narzędzia dydaktyczne						
M-1	Wykład problemowy					
M-2	Wykład informacyjny					
M-3	Anegdota					
M-4	Ćwiczenia laboratoryjne					
M-5	Nauka i ćwiczenia własne					



Sposoby oceny (F - formująca, P - podsumowująca)

S-1	F	Obserwacja aktywności na zajęciach laboratoryjnych
S-2	F	Ocena kompletności rozwiązań problemów postawionych na zajęciach laboratoryjnych
S-3	P	Test zaliczeniowy

Zamierzone efekty kształcenia	Odniesienie do efektów kształcenia dla kierunku studiów	Cel przedmiotu	Treści programowe	Metody nauczania	Sposób oceny
Wiedza					
PESWa_10-09_W01 Zna podstawowe techniki testowania i rodzaje testów i metodykę testowania	PESWa_10-09_W14	C-1	T-W-1 T-W-3 T-W-2 T-W-6	M-1 M-2 M-3	S-3
PESWa_10-09_W02 Ma podstawową wiedzę na temat narzędzi wykorzystywanych w procesie testowania	PESWa_10-09_W14	C-1	T-L-1 T-L-3 T-L-2 T-L-4	M-3 M-4 M-5	S-1 S-2 S-3
Umiejętności					
PESWa_10-09_U01 Umie projektować i tworzyć przypadki testowe z wykorzystaniem narzędzi	PESWa_10-09_U10	C-1	T-L-4 T-W-7 T-W-6	M-1 M-2 M-3 M-4 M-5	S-1 S-2 S-3
PESWa_10-09_U02 Potrafi korzystać z systemu wersjonowania plików	PESWa_10-09_U11	C-1	T-W-2 T-W-5 T-W-3 T-W-6 T-W-4 T-W-7	M-1 M-2	S-3
Inne kompetencje społeczne i personalne					
PESWa_10-09_K01 Rozumie rolę testera w procesie wytwarzania oprogramowania	PESWa_10-09_K02	C-1	T-W-1 T-W-2	M-1 M-2 M-3 M-5	S-3

Efekt	Ocena	Kryterium oceny
Wiedza		
PESWa_10-09_W01	2,0	
	3,0	Zna podstawowe techniki testowania i poziomy testów
	3,5	
	4,0	
	4,5	
	5,0	Poprawnie projektuje przypadki testowe z wykorzystaniem podstawowych technik testowania
PESWa_10-09_W02	2,0	
	3,0	Zna podstawowe narzędzia wykorzystywane w procesie testowania
	3,5	
	4,0	
	4,5	
	5,0	Zna zasady prawidłowego projektowania przypadków testowych
Umiejętności		
PESWa_10-09_U01	2,0	
	3,0	Potrafi obsługiwać narzędzia wykorzystywane w procesie testowania
	3,5	
	4,0	
	4,5	
	5,0	Potrafi prawidłowo zaprojektować przypadki testowe z wykorzystaniem narzędzi
PESWa_10-09_U02	2,0	
	3,0	Potrafi prawidłowo wersjonować pliki
	3,5	
	4,0	
	4,5	
	5,0	Potrafi stworzyć i skonfigurować repozytorium
Inne kompetencje społeczne i personalne		
PESWa_10-09_K01	2,0	
	3,0	Zna podstawy technologii wytwarzania oprogramowania
	3,5	
	4,0	
	4,5	
	5,0	

Literatura podstawowa



Zachodniopomorski Uniwersytet Technologiczny w Szczecinie

Wydział Informatyki

Literatura podstawowa

1. Radosław Smilgin, Zawód tester, PWN

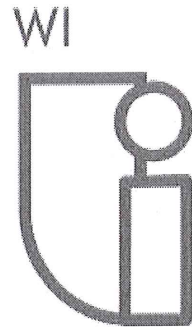
Literatura uzupełniająca

1. Karolina Zmitrowicz, Adam Roman, Testowanie w praktyce. Studium przypadków, PWN



Zachodniopomorski Uniwersytet Technologiczny w Szczecinie

Wydział Informatyki



Kierunek studiów		Programowanie z elementami systemów wbudowanych					
Forma studiów		niestacjonarna	Poziom	podyplomowy			
Obszary studiów		nauki techniczne					
Profil		-					
Moduł							
Przedmiot		Projekt praktyczny - praca zaliczeniowa					
Kod		/SP/PESW/10					
Jednostka prowadząca		Katedra Architektury Komputerów i Teleinformatyki					
ECTS		5,0	ECTS (formy)	5,0			
Forma zaliczenia		egzamin	Język	polski			
Forma dydaktyczna		Kod	Semestr	Godziny	ECTS	Waga	Zaliczenie
projekty		P	2	12	5,0	1,00	egzamin
Nauczyciel odpowiedzialny		Maciaszczyk Radosław (Radoslaw.Maciaszczyk@zut.edu.pl)					
Inni nauczyciele							
Wymagania wstępne							
W-1	Słuchacz zna składnię języka C,C++, potrafi implementować proste algorytmy w języku C, C++, potrafi kompilować programy składające się z wielu plików źródłowych.						
Cele modułu/przedmiotu							
C-1	Słuchacz potrafi stworzyć od podstaw prosty projekt praktyczny						
Treści programowe z podziałem na formy zajęć							Liczba godzin
T-P-1	Przedstawienie projektów do realizacji, analiza funkcjonalności projektów.						2
T-P-2	Implementacja projektów w wybranym środowisku						10
Obciążenie pracą studenta - formy aktywności							Liczba godzin
A-P-1	Uczestnictwo w zajęciach						12
A-P-2	Konsultacje z opiekunem projektu						12
A-P-3	Samodzielna praca nad projektem						100
Metody nauczania / narzędzia dydaktyczne							
M-1	Klasyczna metoda problemowa						
M-2	Metoda projektów						
M-3	Nauka i ćwiczenia własne						
Sposoby oceny (F - formująca, P - podsumowująca)							
S-1	F	Obserwacja aktywności na zajęciach					
S-2	F	Ocena kompletności rozwiązań problemów postawionych w projekcie					
S-3	P	Prezentacja końcowa					
Zamierzone efekty kształcenia		Odniesienie do efektów kształcenia dla kierunku studiów	Cel przedmiotu	Treści programowe	Metody nauczania	Sposób oceny	
Wiedza							
PESWa_10_10_W01 Potrafi wykorzystać wiedzę zdobytą w trakcie studiów w procesie tworzenia od podstaw projektu		PESWa_10_W03 PESWa_10_W04 PESWa_10_W05 PESWa_10_W06 PESWa_10_W07 PESWa_10_W08 PESWa_10_W09 PESWa_10_W10 PESWa_10_W11 PESWa_10_W12 PESWa_10_W13 PESWa_10_W14	C-1	T-P-1 T-P-2	M-1 M-2 M-3	S-1 S-2 S-3	
Umiejętności							



Wydział Informatyki

PESWa_10-10_U01 Umie stworzyć program na podstawie zadanego zadania projektowego	PESWa_10-U02 PESWa_10-U03 PESWa_10-U04 PESWa_10-U05 PESWa_10-U06 PESWa_10-U07 PESWa_10-U08 PESWa_10-U10 PESWa_10-U11	C-1	T-P-1 T-P-2	M-1 M-2 M-3	S-1 S-2 S-3
---	--	-----	-------------	-------------------	-------------------

Inne kompetencje społeczne i personalne

PESWa_10-10_K01 Rozumie rolę programisty w procesie projektowania, wykonania, testowania oprogramowania.	PESWa_10-K01 PESWa_10-K02	C-1	T-P-1 T-P-2	M-1 M-2 M-3	S-2 S-3
---	------------------------------	-----	-------------	-------------------	------------

Efekt	Ocena	Kryterium oceny			
-------	-------	-----------------	--	--	--

Wiedza

PESWa_10-10_W01	2,0	
	3,0	Potrafi wskazać rozwiązania techniczne do realizacji problemu postawionego w projekcie
	3,5	
	4,0	
	4,5	
	5,0	

Umiejętności

PESWa_10-10_U01	2,0	
	3,0	Potrafi zaimplementować program realizujący w sposób podstawowy problem postawiony w projekcie
	3,5	
	4,0	
	4,5	
	5,0	

Inne kompetencje społeczne i personalne

PESWa_10-10_K01	2,0	
	3,0	Potrafi wskazać zagrożenia jakie mogą wystąpić w danym projekcie wynikające z błędnej implementacji.
	3,5	
	4,0	
	4,5	
	5,0	

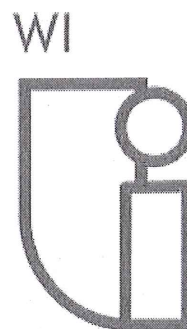
Literatura podstawowa

1. Robert C. Martin, Czysty kod. Podręcznik dobrego programisty, Helion, 2010



Zachodniopomorski Uniwersytet Technologiczny w Szczecinie

Wydział Informatyki



Kierunek studiów		Programowanie z elementami systemów wbudowanych					
Forma studiów		niestacjonarna	Poziom	podyplomowy			
Obszary studiów		nauki techniczne					
Profil		-					
Moduł							
Przedmiot		Programowanie mikrokomputerów jednopłytkowych					
Kod		/SP/PESW/11					
Jednostka prowadząca		Katedra Architektury Komputerów i Teleinformatyki					
ECTS		2,5	ECTS (formy)	2,5			
Forma zaliczenia		zaliczenie	Język	polski			
Forma dydaktyczna	Kod	Semestr	Godziny	ECTS	Waga	Zaliczenie	
wykłady	W	2	4	0,6	0,25	zaliczenie	
laboratoria	L	2	12	1,9	0,75	zaliczenie	
Nauczyciel odpowiedzialny		Maciaszczyk Radosław (Radoslaw.Maciaszczyk@zut.edu.pl)					
Inni nauczyciele							
Wymagania wstępne							
W-1	Podstawy język C i C++						
W-2	Obsługa systemu Linux						
Cele modułu/przedmiotu							
C-1	Zapoznanie z architekturą mikrokomputerów oraz z obsługą portów wejściowych i wyjściowych.						
Treści programowe z podziałem na formy zajęć						Liczba godzin	
T-W-1	Idea mikrokomputerów i ich zastosowania, architektura, możliwości połączeniowe, obsługiwane systemy operacyjne.					2	
T-W-2	Obsługa interfejsów wbudowanych w mikrokontroler.					2	
T-L-1	Instalacja i konfiguracja systemu operacyjnego na mikrokomputerze, sterowanie mikrokomputerem na przykładzie platformy Raspberry Pi					2	
T-L-2	Obsługa GPIO.					2	
T-L-3	Obsługa komunikacji szeregowej.					4	
T-L-4	Realizacja projektów zadanych przez prowadzącego					4	
Obciążenie pracą studenta - formy aktywności						Liczba godzin	
A-W-1	Uczestnictwo w wykładach					4	
A-W-2	Samodzielnie studiowanie zalecanej literatury i analiza materiału z wykładów					12	
A-W-3	Konsultacje					2	
A-L-1	Uczestnictwo w zajęciach					12	
A-L-2	Samodzielne rozwiązywanie zadań i problemów postawionych na laboratoriach					24	
A-L-3	Przygotowanie do zajęć					12	
A-L-4	Konsultacje					4	
A-L-5	Przygotowanie prezentacji projektu					4	
Metody nauczania / narzędzia dydaktyczne							
M-1	Wykład informacyjny						
M-2	Ćwiczenia laboratoryjne						
Sposoby oceny (F - formująca, P - podsumowująca)							
S-1	F	Ocena kompletności rozwiązań problemów postawionych na zajęciach laboratoryjnych.					
S-2	F	Ocena stopnia realizacji projektu.					
Zamierzone efekty kształcenia			Odniesienie do efektów kształcenia dla kierunku studiów	Cel przedmiotu	Treści programowe	Metody nauczania	Sposób oceny
Wiedza							
PESWa_10-11_W01 Zna architekturę mikrokomputerów			PESWa_10-W15	C-1	T-L-1 T-W-1	M-1 M-2	S-1 S-2



Wydział Informatyki

PESWa_10-11_W02 Zna metody komunikacji z urządzeniami peryferyjnymi.	PESWa_10-W16	C-1	T-L-1 T-L-2 T-L-3	T-L-4 T-W-1 T-W-2	M-1 M-2	S-1 S-2
---	--------------	-----	-------------------------	-------------------------	------------	------------

Umiejętności

PESWa_10-11_U01 Potrafi napisać program komunikujący się z urządzeniami peryferyjnymi	PESWa_10-U12 PESWa_10-U13	C-1	T-L-1 T-L-2 T-L-3	T-L-4 T-W-1 T-W-2	M-1 M-2	S-1 S-2
--	------------------------------	-----	-------------------------	-------------------------	------------	------------

Inne kompetencje społeczne i personalne

Efekt	Ocena	Kryterium oceny
-------	-------	-----------------

Wiedza

PESWa_10-11_W01	2,0	
	3,0	Potrafi określić funkcje poszczególnych elementów architektury mikrokontrolerów
	3,5	
	4,0	
	4,5	
	5,0	
PESWa_10-11_W02	2,0	
	3,0	Potrafi określić w jaki sposób należy skomunikować się z wybranymi urządzeniami zewnętrznymi
	3,5	
	4,0	
	4,5	
	5,0	

Umiejętności

PESWa_10-11_U01	2,0	
	3,0	Potrafi napisać program, który wymienia danem z wybranym urządzeniem peryferyjnym.
	3,5	
	4,0	
	4,5	
	5,0	

Inne kompetencje społeczne i personalne

Literatura podstawowa

1. Simon Monk, Elektronika z wykorzystaniem Arduino i Rapsberry Pi. Receptury, Helion, 2018
2. Richardson Matt, Wallace Shawn, Wprowadzenie do Raspberry Pi. Wydanie II, Promise, 2016

Literatura uzupełniająca

1. Dominique Guinard, Vlad Trifa, Internet rzeczy. Budowa sieci z wykorzystaniem technologii webowych i Raspberry Pi, Helion, 2017

**Wewnętrzny system zapewniania jakości kształcenia
na studiach podyplomowych pn.
Programowanie z elementami systemów wbudowanych**

Zastosowanie się do bezwzględnych potrzeb i dążeń uczestników studiów podyplomowych, ich pracodawców oraz ogółu społeczeństwa odbywać się będzie przy zastosowaniu i wdrożeniu zasad wynikających z Zarządzenia Rektora ZUT w Szczecinie w sprawie podstaw funkcjonowania Wewnętrznego Systemu Zapewniania Jakości Kształcenia oraz wytycznych dotyczących realizacji oceny jakości w obszarach tego systemu:

- weryfikowanie realizacji osiągniętych efektów kształcenia,
- ocenę i analizę procesu kształcenia,
- analizę opinii uczestników studiów podyplomowych z wykorzystaniem ankiet.

Akty prawne Zachodniopomorskiego Uniwersytetu Technologicznego w Szczecinie związane z Systemem Zapewniania Jakości Kształcenia:

- Uchwała Senatu ZUT w Szczecinie ws. wewnętrznego systemu zapewniania jakości kształcenia;
- Uchwała Senatu ZUT w Szczecinie ws. wprowadzenia polityki jakości kształcenia;
- Zarządzenie Rektora ZUT w Szczecinie ws. podstaw funkcjonowania Wewnętrznego Systemu Zapewniania jakości Kształcenia oraz wytycznych do realizacji oceny jakości w obszarach działania tego systemu;
- Zarządzenie Rektora ZUT w Szczecinie ws. wprowadzenia regulaminu uczelnianej i wydziałowej komisji ds. jakości kształcenia.