



### KARTA PRZEDMIOTU

Kod przedmiotu	studia stacjonarne:	<b>Z-ZIP2-U-231</b>
	studia niestacjonarne:	<b>Z-ZIPN2-U-231</b>
Nazwa przedmiotu	<b>Zaawansowane techniki programowania</b>	
Nazwa przedmiotu w języku angielskim	<b>Advanced Programming</b>	
Obowiązuje od roku akademickiego	<b>2019/2020</b>	

### USYTUOWANIE PRZEDMIOTU W SYSTEMIE STUDIÓW

Kierunek studiów	<b>ZARZĄDZANIE I INŻYNIERIA PRODUKCJI</b>
Poziom kształcenia	<b>II stopień</b>
Profil studiów	<b>Ogólnoakademicki</b>
Forma i tryb prowadzenia studiów	<b>Studia stacjonarne</b>
Zakres	<b>Informatyka w zarządzaniu i modelowaniu</b>
Jednostka prowadząca przedmiot	<b>Katedra Technologii Informatycznych</b>
Koordinator przedmiotu	<b>dr inż. Sławomir Koczubiej</b>
Zatwierdził	<b>dr hab. inż. Artur Bartosik, prof. PŚk</b>

### OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

Przynależność do grupy/bloku przedmiotów	<b>Przedmiot specjalnościowy</b>	
Status przedmiotu	<b>Obowiązkowy</b>	
Język prowadzenia zajęć	<b>Polski</b>	
Usytuowanie w planie studiów - semestr	studia stacjonarne	<b>Semestr II</b>
	studia niestacjonarne	<b>-</b>
Wymagania wstępne	<b>Brak</b>	
Egzamin (TAK/NIE)	<b>TAK</b>	
Liczba punktów ECTS	<b>3</b>	

Forma prowadzenia zajęć		wykład	ćwiczenia	laboratorium	projekt	inne
Liczba godzin w semestrze	studia stacjonarne:	<b>20</b>		<b>30</b>		
	studia niestacjonarne:	<b>-</b>	<b>-</b>	<b>-</b>	<b>-</b>	<b>-</b>

## EFEKTY UCZENIA SIĘ

Kategoria	Symbol efektu	Efekty uczenia się	Odniesienie do efektów kierunkowych
Wiedza	W01	Poznanie zaawansowanych modeli programowania obiektowego. Ma wiedzę dotyczącą różnych paradygmatów programowania.	ZIP2_W04
	W02	Posiada wiedzę dotyczącą budowania aplikacji wieloplatformowych.	ZIP2_W04
	W03	Student ma wiedzę dotyczącą modeli programowania i doboru ich do rozwiązania określonego zadania programistycznego.	ZIP2_W04
Umiejętności	U01	Student posiada umiejętność konstruowania algorytmów z wykorzystaniem zaawansowanych struktur danych służących rozwiązaniu różnych zadań programistycznych.	ZIP2_U11
	U02	Potrafi ocenić nowe technologie nowe narzędzia programistyczne, diagnostyczne i dokonać ich wyboru realizacji zadanych problemów.	ZIP2_U12
	U03	Posiada umiejętność wykorzystywania zasobów programistycznych i wykorzystać właściwe narzędzia programistyczne pozwalające na sprawne realizowanie projektów.	ZIP2_U11
Kompetencje społeczne	K01	Zdolność do kierowania zespołem ludzkim z uwzględnieniem czynników psychologicznych i społecznych członków zespołu.	ZIP2_K03

## TREŚCI PROGRAMOWE

Forma zajęć	Treści programowe
wykład	<ol style="list-style-type: none"><li>1. Metodologia budowania dużych projektów, kontrola wersji, zarządzanie projektem. Metodyka budowania testów. Biblioteki testujące. Optymalizacja kodu.</li><li>2. Interfejsy i klasy ogólne. Metody anonimowe. Wielowątkowość, metody synchronizacji.</li><li>3. Zaawansowane struktury danych i algorytmy. Wyrażenia regularne.</li><li>4. Zaawansowane programowanie obiektowe. Wzorce projektowe.</li></ol>
laboratorium	<ol style="list-style-type: none"><li>1. Zarządzanie projektem, kontrola wersji i zarządzanie kodem aplikacji. Pluginy, programowanie dynamiczne.</li><li>2. Serializacja i klonowanie. Aplikacje wielowątkowe, metody synchronizacji wątków. Programowanie asynchroniczne.</li><li>3. Zaawansowane biblioteki struktur danych. Aplikacje wieloplatformowe.</li><li>4. Interfejsy i klasy ogólne, uniwersalne implementacje algorytmów. Kreatywne, strukturalne i czynnościowe wzorce projektowe.</li></ol>

## METODY WERYFIKACJI EFEKTÓW UCZENIA SIĘ

Symbol efektu	Metody sprawdzania efektów kształcenia (zaznaczyć X)					
	Egzamin ustny	Egzamin pisemny	Kolokwium	Projekt	Sprawozdanie	Inne
W01		X	X			
W02		X	X			
W03		X	X			
U01		X	X		X	
U02		X	X		X	
U03		X	X		X	
K01					X	

## FORMA I WARUNKI ZALICZENIA

Forma zajęć	Forma zaliczenia	Warunki zaliczenia
wykład	egzamin	Uzyskanie co najmniej 50% punktów z egzaminu, komentarze na wykładach.
laboratorium	zaliczenie z oceną	Uzyskanie co najmniej 50% z kolokwίων, aktywność na zajęciach laboratoryjnych.

## NAKŁAD PRACY STUDENTA

Bilans punktów ECTS												
Lp.	Rodzaj aktywności	Obciążenie studenta										Jednostka
		studia stacjonarne					studia niestacjonarne					
		W	C	L	P	S	W	C	L	P	S	
1.	Udział w zajęciach zgodnie z planem studiów	20		30			-	-	-	-	-	h
2.	Inne (konsultacje, egzamin)	4		2			-	-	-	-	-	h
3.	<b>Razem przy bezpośrednim udziale nauczyciela akademickiego</b>	<b>56</b>					<b>-</b>					h
4.	<b>Liczba punktów ECTS, którą student uzyskuje przy bezpośrednim udziale nauczyciela akademickiego</b>	<b>2,2</b>					<b>-</b>					ECTS
5.	<b>Liczba godzin samodzielnej pracy studenta</b>	<b>19</b>					<b>-</b>					h
6.	<b>Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy</b>	<b>0,8</b>					<b>-</b>					ECTS
7.	<b>Nakład pracy związany z zajęciami o charakterze praktycznym</b>	<b>45</b>					<b>-</b>					h
8.	<b>Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym</b>	<b>1,8</b>					<b>-</b>					ECTS
9.	<b>Sumaryczne obciążenie pracą studenta</b>	<b>75</b>					<b>-</b>					h
10.	<b>Punkty ECTS za moduł</b> <i>1 punkt ECTS=25 godzin obciążenia studenta</i>	<b>3</b>										ECTS

## LITERATURA

1. Galowicz J. (2018), *C++ STL. Receptury*, Wydawnictwo Helion, Gliwice.
2. Ezust A., Ezust P. (2014), *C++ i Qt. Wprowadzenie do wzorców projektowych*, Helion, Gliwice.
3. Gamma E., Helm R., Johnson R., Vlissides J. (2010), *Wzorce projektowe. Elementy oprogramowania obiektowego wielokrotnego użytku*. Wydawnictwo Helion, Gliwice.