



KARTA MODUŁU / KARTA PRZEDMIOTU

Kod modułu	
Nazwa modułu	Programowanie w języku C++ 2
Nazwa modułu w języku angielskim	Programming in C++ 2
Obowiązuje od roku akademickiego	2017/2018

A. USYTUOWANIE MODUŁU W SYSTEMIE STUDIÓW

Kierunek studiów	Informatyka
Poziom kształcenia	I stopień
Profil studiów	ogólnoakademicki
Forma i tryb prowadzenia studiów	stacjonarne
Specjalność	Teleinformatyka
Jednostka prowadząca moduł	Zakład Informatyki
Koordynator modułu	dr hab. inż. Roman Stanisław Deniziak, prof. PŚk mgr inż. Paweł Pięta
Zatwierdził:	Dziekan WEAiI dr hab. inż. Antoni Różowicz, prof. PŚk

B. OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

Przynależność do grupy/bloku przedmiotów	przedmiot specjalnościowy
Status modułu	obieralny
Język prowadzenia zajęć	polski
Usytuowanie modułu w planie studiów - semestr	semestr VII
Usytuowanie realizacji przedmiotu w roku akademickim	semestr zimowy
Wymagania wstępne	Programowanie w języku C++ 1
Egzamin	nie
Liczba punktów ECTS	7

Forma prowadzenia zajęć	wykład	ćwiczenia	laboratorium	projekt	inne
w semestrze	15		30	30	



C. EFEKTY KSZTAŁCENIA I METODY SPRAWDZANIA EFEKTÓW KSZTAŁCENIA

Cel modułu	Celem modułu jest zdobycie wiedzy oraz praktycznych umiejętności obejmujących bardziej zaawansowane zagadnienia związane z programowaniem w języku C++, które dotyczą szablonów funkcji i klas oraz standardowej biblioteki szablonów (STL).
-------------------	--

Symbol efektu	Efekty kształcenia	Forma prowadzenia zajęć (w/ć/l/p/inne)	odniesienie do efektów kierunkowych	odniesienie do efektów obszarowych
W_01	Wiedza na temat zagadnień związanych z wykorzystaniem szablonów funkcji i klas w języku C++.	W	K_W11	T1A_W03 T1A_W04 T1A_W07
W_02	Wiedza na temat zagadnień związanych z wykorzystaniem standardowej biblioteki szablonów (STL) języka C++.	W	K_W11 K_W15	T1A_W03 T1A_W04 T1A_W07
W_03	Wiedza na temat sposobów rozwiązywania problemów programistycznych z zastosowaniem szablonów funkcji i klas oraz standardowej biblioteki szablonów (STL) języka C++.	W	K_W07 K_W11 K_W15	T1A_W03 T1A_W04 T1A_W07
W_04	Wiedza na temat zagadnień związanych z zaawansowanymi operacjami wejścia/wyjścia w języku C++ (STL I/O).	W	K_W11	T1A_W03 T1A_W04 T1A_W07
U_01	Umiejętność tworzenia programów w języku C++ z wykorzystaniem szablonów funkcji i klas.	L, P	K_U17	T1A_U08 T1A_U09 T1A_U13 T1A_U15 T1A_U16
U_02	Umiejętność tworzenia programów w języku C++ z wykorzystaniem standardowej biblioteki szablonów (STL).	L, P	K_U17	T1A_U08 T1A_U09 T1A_U13 T1A_U15 T1A_U16
U_03	Umiejętność rozwiązywania problemów programistycznych z zastosowaniem szablonów funkcji i klas oraz standardowej biblioteki szablonów (STL) języka C++.	L, P	K_U02 K_U13 K_U17	T1A_U02 T1A_U08 T1A_U09 T1A_U13 T1A_U14 T1A_U15 T1A_U16
U_04	Umiejętność tworzenia programów w języku C++ z wykorzystaniem zaawansowanych operacji wejścia/wyjścia (STL I/O).	L, P	K_U17	T1A_U08 T1A_U09 T1A_U13 T1A_U15 T1A_U16
U_05	Umiejętność opracowania dokumentacji i sprawozdania dotyczących rozwiązanych problemów programistycznych.	P	K_U03	T1A_U03
K_01	Kompetencja do pracy w zespole podczas rozwiązywania problemów programistycznych i umiejętność ich podziału na mniejsze zadania.	P	K_K03	T1A_K03 T1A_K04



Treści kształcenia:

1. Treści kształcenia w zakresie wykładu

Nr wykładu	Treści kształcenia	Odniesienie do efektów kształcenia dla modułu
1	Wstęp do standardowej biblioteki szablonów (STL). Kontenery vector, deque i list. Iteratory.	W_01 W_02
2	Podstawowe metody kontenerów. Adaptery kontenerów.	W_01 W_02 W_03
3	Kontenery set, multiset, map i multimap.	W_01 W_02
4	Algorytmy w STL: operacje niemodyfikujące i operacje modyfikujące.	W_01 W_02 W_03
5	Algorytmy w STL: operacje sortujące, operacje na posortowanych ciągach, operacje min/max. Obiekty funkcyjne.	W_01 W_02 W_03
6	Zaawansowane operacje wejścia/wyjścia (STL I/O).	W_01 W_02 W_04
7	Szablony funkcji i klas.	W_01 W_03

2. Treści kształcenia w zakresie ćwiczeń – nie dotyczy

Nr zajęć ćwic.	Treści kształcenia	Odniesienie do efektów kształcenia dla modułu

3. Treści kształcenia w zakresie zadań laboratoryjnych

Nr zajęć lab.	Treści kształcenia	Odniesienie do efektów kształcenia dla modułu
1	Wstęp do standardowej biblioteki szablonów (STL).	U_01 U_02
2	Kontener vector.	U_01 U_02
3	Kontenery deque i list.	U_01 U_02
4	Iteratory.	U_01 U_02
5	Podstawowe metody kontenerów.	U_01 U_02
6	Adaptery kontenerów.	U_01 U_02 U_03
7	Kontenery set i multiset.	U_01 U_02
8	Kontenery map i multimap.	U_01 U_02
9	Algorytmy w STL: operacje niemodyfikujące.	U_01



		U_02 U_03
10	Algorytmy w STL: operacje modyfikujące.	U_01 U_02 U_03
11	Algorytmy w STL: operacje sortujące.	U_01 U_02 U_03
12	Algorytmy w STL: operacje na posortowanych ciągach i operacje min/max.	U_01 U_02 U_03
13	Obiekty funkcyjne.	U_01 U_02 U_03
14	Zaawansowane operacje wejścia/wyjścia (STL I/O).	U_01 U_02 U_04
15	Szablony funkcji i klas.	U_01 U_03

4. Charakterystyka zadań projektowych

Zadania projektowe będą polegały na zaimplementowaniu aplikacji w języku C++ z zastosowaniem paradygmatu programowania obiektowego. Program powinien wykorzystywać szablony funkcji i klas oraz standardową bibliotekę szablonów (STL). Aplikacja powinna posiadać intuicyjny interfejs graficzny. Projekty będą realizowane w zespołach dwuosobowych.

Metody sprawdzania efektów kształcenia

Symbol efektu	Metody sprawdzania efektów kształcenia (sposób sprawdzenia, w tym dla umiejętności – odwołanie do konkretnych zadań projektowych, laboratoryjnych, itp.)
W_01	Kolokwium z wiedzy teoretycznej.
W_02	Kolokwium z wiedzy teoretycznej.
W_03	Kolokwium z wiedzy teoretycznej.
W_04	Kolokwium z wiedzy teoretycznej.
U_01	Zadania laboratoryjne, kolokwia zaliczeniowe, zadania projektowe.
U_02	Zadania laboratoryjne, kolokwia zaliczeniowe, zadania projektowe.
U_03	Zadania laboratoryjne, kolokwia zaliczeniowe, zadania projektowe.
U_04	Zadania laboratoryjne, kolokwia zaliczeniowe, zadania projektowe.
U_05	Sprawozdanie i dokumentacja techniczna projektu.
K_01	Zadania projektowe – praca w zespołach.



D. NAKŁAD PRACY STUDENTA

Bilans punktów ECTS	
Rodzaj aktywności	obciążenie studenta
Udział w wykładach	15
Udział w ćwiczeniach	
Udział w laboratoriach	30
Udział w konsultacjach (2-3 razy w semestrze)	15
Udział w zajęciach projektowych	30
Konsultacje projektowe	15
Udział w egzaminie	
Liczba godzin realizowanych przy bezpośrednim udziale nauczyciela akademickiego	105 (suma)
Liczba punktów ECTS, którą student uzyskuje na zajęciach wymagających bezpośredniego udziału nauczyciela akademickiego (1 punkt ECTS=25-30 godzin obciążenia studenta)	4
Samodzielne studiowanie tematyki wykładów	15
Samodzielne przygotowanie się do ćwiczeń	
Samodzielne przygotowanie się do kolokwiów	15
Samodzielne przygotowanie się do laboratoriów	15
Wykonanie sprawozdań	10
Przygotowanie do kolokwium końcowego z laboratorium	10
Wykonanie projektu lub dokumentacji	15
Przygotowanie do egzaminu	
Liczba godzin samodzielnej pracy studenta	80 (suma)
Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy (1 punkt ECTS=25-30 godzin obciążenia studenta)	3
Summaryczne obciążenie pracą studenta	185
Punkty ECTS za moduł 1 punkt ECTS=25-30 godzin obciążenia studenta	7

E. LITERATURA

Wykaz literatury	Materiały zawarte na platformie NetAcad udostępniane studentom podczas zajęć dydaktycznych.
Witryna WWW modułu/przedmiotu	www.netacad.com