

KARTA MODUŁU / KARTA PRZEDMIOTU

Kod modułu	
Nazwa modułu	Język programowania C
Nazwa modułu w języku angielskim	C Programming Language
Obowiązuje od roku akademickiego	2013/2014

A. USYTUOWANIE MODUŁU W SYSTEMIE STUDIÓW

Kierunek studiów	Mechanika i Budowa Maszyn
Poziom kształcenia	II stopień
Profil studiów	ogólnoakademicki
Forma i tryb prowadzenia studiów	studia stacjonarne
Specjalność	Systemy CAD/CAE
Jednostka prowadząca moduł	Katedra Automatyki i Robotyki
Koordynator modułu	Dr hab. inż. Jerzy Stamirowski
Zatwierdził:	

B. OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

Przynależność do grupy/bloku przedmiotów	przedmiot podstawowy
Status modułu	przedmiot obowiązkowy
Język prowadzenia zajęć	polski
Usytuowanie modułu w planie studiów - semestr	pierwszy
Usytuowanie realizacji przedmiotu w roku akademickim	semestr letni
Wymagania wstępne	podstawy informatyki
Egzamin	nie
Liczba punktów ECTS	3

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

C. EFEKTY KSZTAŁCENIA I METODY SPRAWDZANIA EFEKTÓW KSZTAŁCENIA

Cel modułu	Celem przedmiotu jest nauczenie studentów programowania w języku C. Student powinien posiadać umiejętność pisania i uruchamiania programów w języku C dla obliczeniowych zadań numerycznych oraz znać rolę języka C w budowie oprogramowania CAD/CAE.
-------------------	---

Symbol efektu	Efekty kształcenia	Forma prowadzenia zajęć (w/ć/l/p/inne)	odniesienie do efektów kierunkowych	odniesienie do efektów obszarowych
W_01	Student zna i rozumie metody przechowywania i używania prostych danych oraz operatory i instrukcje sterujące. Wiedza ta pozwoli mu pisać programy dla prostych zadań obliczeniowych.	wykład laboratorium	KS_W01_ CAD/CAE	T2A_W04 InzA_W01
W_02	Student zna i rozumie proste struktury danych (wektory, tablice, macierze, listy itp.), wskaźniki i referencje, operacje na bitach, funkcje i ich użycie. Wiedza ta pozwoli mu pisać programy dla zadań obliczeniowych o średnim poziomie złożoności.	wykład laboratorium	KS_W01_ CAD/CAE	T2A_W04 InzA_W01
W_03	Student zna złożone struktury danych (struktury listowe, stosy i kolejki, grafy) i ich użycie w programach. Wiedza ta pozwoli mu programować zadania algorytmiczne w języku C. Student zna operacje na strumieniach wejścia i wyjścia w szczególności na plikach i ich indywiduizację.	wykład laboratorium	KS_W01_ CAD/CAE	T2A_W04 InzA_W01
W_04	Student zna metody programowania zadań algorytmicznych.	wykład, laboratorium	KS_W01_ CAD/CAE	T2A_W04 InzA_W01
W_05	Student ma wiedzę na temat tworzenia z użyciem bibliotek graficznych wykresów dla funkcji i danych oraz wiedzę na temat budowy prostych graficznych interfejsów użytkownika. Wiedza ta pozwoli mu budować dość zaawansowane aplikacje w języku C.	wykład laboratorium	KS_W01_ CAD/CAE	T2A_W04 InzA_W01
U_01	Student potrafi napisać w języku C. program dla prostych zadań obliczeniowych	Laboratorium	KU_11 KS_U01_ CAD/CAE	T2A_U07 T2A_U08 T2A_U09 InzA_U08 InzA_U08
U_02	Student potrafi napisać w języku C program dla zaawansowanych algorytmicznych zadań obliczeniowych.	Laboratorium	KU_11 KS_U01_ CAD/CAE	T2A_U07 T2A_U08 T2A_U09 InzA_U08 InzA_U08

U_03	Student potrafi posługiwać się programami w języku C przy realizacji typowych zadań inżynierskich	Laboratorium	KU_11 KS_U01_ CAD/CAE	T2A_U07 T2A_U08 T2A_U09 InżA_U08 InżA_U08
K_01	Student ma świadomość odpowiedzialności za pracę w obszarze tworzenia oprogramowania.	Wykład laboratorium	KK_01	T2A_K01 T2A_K03
K_02	Student rozumie potrzebę dokształcania się w dziedzinie programowania komputerów	Wykład laboratorium	KK_01	T2A_K01 T2A_K03

Treści kształcenia:

1. Treści kształcenia w zakresie wykładu

Nr wykładu	Treści kształcenia	Odniesienie do efektów kształcenia dla modułu
1	Środowisko programowania. Programowanie niskiego poziomu w języku C: stałe i zmienne (typy i nazwy), przestrzenie nazw, operatory i wyrażenia, instrukcje sterujące, funkcje. Pisanie programów dla prostych zadań obliczeniowych.	W_01
2	Dane i algorytmy w języku C. Struktury danych, (wektory, tablice, struktury i unie, macierze) i ich użycie przy pisaniu programów dla zadań obliczeniowych.	W_02
3	Dane i algorytmy w języku C. Wskaźniki i referencje. Złożone struktury danych (struktury listowe, kolejki i stosy, drzewa). Użycie funkcji i struktur danych przy pisaniu programów dla zadań obliczeniowych.	W_02 W_03
4	Strumienie wejścia i wyjścia i ich indywidualizacja. Operacje na plikach (Otwieranie, odczytywanie, zapisywanie, wczytywanie pliku strukturalnego) Formatowanie danych wyjściowych. Konstruowanie programu i kończenie programu.	W_04
5	Programowanie obliczeniowych zadań algorytmicznych.	W_02 W_03 W_04
6	Biblioteki graficzne. Użycie przy rysowaniu wykresów dla funkcji i danych. Graficzny interfejs użytkownika dla prostej aplikacji.	W_05
7	Zastosowanie w zadaniach inżynierskich programów i aplikacji w języku C . Rozwój języka C. Porównanie z językiem C++	W_04 K_01 K_02

2. Treści kształcenia w zakresie zadań laboratoryjnych

Nr zajęć lab.	Treści kształcenia	Odniesienie do efektów kształcenia dla modułu
1.	Kompilatory. Środowisko programowania.	W_01
2.	Poprawne pisanie programu i kończenie programu. Zmienne	W_01
3.	Programy dla prostych zadań obliczeniowych. Użycie funkcji	W_02 U_01

4.	Proste struktury danych (wektory, tablice, struktury i unie, macierze) i ich użycie	W_02
5.	Złożone struktury danych (struktury listowe, stosy i kolejki, kontenery i iteratory)	W_02 W_03
6.	Złożone struktury danych (drzewa, drzewa wyważone, drzewa decyzyjne).	W_03
7.	Sortowanie i wyszukiwanie. kodowanie mieszające.	W_03 W_04
8.	Grafy	W_03 W_04
9.	Działania na liczbach. Biblioteka Matrix. Rozwiązywanie równań liniowych, standardowe funkcje matematyczne, liczby losowe, liczby zespolone.	W_03 W_04 U_02
10.	Wybrane algorytmy. Najkrótsze ścieżki, operacje na macierzach.	W_04 U_02
11.	Wybrane algorytmy. Wyszukiwanie wzorca, algorytmy aproksymacyjne.	W_04 U_02
12.	Model graficzny. Biblioteka GUI. Klasy graficzne.	W_05 U_03
13.	Graficzne przedstawienie funkcji i danych. Wykresy.	W_05 U_03
14.	Graficzne interfejsy użytkownika.	W_05 U_03
15.	Błędy. Testowanie.	K_01

3. Charakterystyka zadań projektowych

4. Charakterystyka zadań w ramach innych typów zajęć dydaktycznych

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 do W_04	Końcowy sprawdzian pisemny Pytania i programy z pełnego zakresu wiedzy objętego programem wykładu. Ocena studenta uzależniona od ilości zdobytych punktów.
W_01 do W_04	Sprawdzian pisemny cząstkowy. Ocena uwzględniana przy ocenie końcowej.
U_01	Sprawdzian. Samodzielne zadanie programistyczne.
U_02	Sprawdzian. Samodzielne zadanie programistyczne.
K_01 K_02	Obserwacja postawy studenta podczas zajęć dydaktycznych. Dyskusja w czasie laboratorium

D. NAKŁAD PRACY STUDENTA

Bilans punktów ECTS		
	Rodzaj aktywności	obciążenie studenta
1	Udział w wykładach	15h
3	Udział w laboratoriach	30h
4	Udział w konsultacjach (2-3 razy w semestrze)	5h

5	Udział w zajęciach projektowych	
6	Konsultacje projektowe	
7	Udział w sprawdzianie końcowym z wykładów	5h
8		
9	Liczba godzin realizowanych przy bezpośrednim udziale nauczyciela akademickiego	55
10	Liczba punktów ECTS, którą student uzyskuje na zajęciach wymagających bezpośredniego udziału nauczyciela akademickiego	2,0 ECTS
11	Samodzielne studiowanie tematyki wykładów	5h
12	Samodzielne przygotowanie się do ćwiczeń	
13	Samodzielne przygotowanie się do kolokwium	
14	Samodzielne przygotowanie się do laboratoriów	5h
15	Wykonanie sprawozdań	10h
15	Przygotowanie do kolokwium końcowego z laboratorium	
17	Wykonanie projektu lub dokumentacji	
18	Przygotowanie do sprawdzianu końcowego z wykładów	10h
19		
20	Liczba godzin samodzielnej pracy studenta	30h
21	Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy	1,0 ECTS
22	Sumaryczne obciążenie pracą studenta	85h
23	Punkty ECTS za moduł	3,0 ECTS
24	Nakład pracy związany z zajęciami o charakterze praktycznym	
25	Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym	

E. LITERATURA

Wykaz literatury	1. Stroustrup Bjarne. <i>Programowanie. Teoria i praktyka z wykorzystaniem C++</i>. Wydawnictwo Helion, Gliwice 2010. 2. Thomas H.Cormen, Charles E.Leiserson, Roland L.Rivest. <i>Wprowadzenie do algorytmów</i>. WNT, 1998. 3. Baron B. <i>Metody numeryczne w C++Builder</i>, Wydawnictwo Helion, Gliwice 2004. 4. Grębosz J. <i>Symfonia C++ standard tom1 i tom2</i>, Wydawnictwo Edition 2000, Kraków 2008. 5. Deitel H.M. Deitel P.J. <i>C++ How to program Third Edition</i> Wydawnictwo Prentice Hall, 2001 6. Porębski W. <i>Język C++ Standard ISO w praktyce</i>. Wydawnictwo Naukowo-Techniczne PWN, Warszawa 2008.
Witryna WWW modułu/przedmiotu	