

(pieczęć wydziału)

KARTA PRZEDMIOTU

Z1-PU7

WYDANIE N1

Strona 1 z 3

1. Nazwa przedmiotu: PROGRAMOWANIE KOMPUTERÓW		2. Kod przedmiotu: PK		
3. Karta przedmiotu ważna od roku akademickiego: 2013/2014				
4. Forma kształcenia: studia pierwszego stopnia				
5. Forma studiów: studia niestacjonarne zaoczne				
6. Kierunek studiów: INFORMATYKA (RAU)				
7. Profil studiów: ogólnoakademicki				
8. Specjalność: -				
9. Semestr: II, III				
10. Jednostka prowadząca przedmiot: Wydział Automatyki, Elektroniki i Informatyki				
11. Prowadzący przedmiot: dr inż. Karolina Nurzyńska				
12. Przynależność do grupy przedmiotów: przedmioty wspólne				
13. Status przedmiotu: obowiązkowy				
14. Język prowadzenia zajęć: polski				
15. Przedmioty wprowadzające oraz wymagania wstępne: Przedmiot: Podstawy programowania komputerów.				
16. Cel przedmiotu: Przedmiot ma przekazać wiedzę niezbędną do zrozumienia, projektowania i pisania programów oraz umiejętność implementacji oprogramowania na przykładzie języków C i C++. Celem przedmiotu jest stworzenie mocnych podstaw zarówno teoretycznych, jak i praktycznych w zakresie tworzenia oprogramowania. Laboratorium umożliwia zastosowanie w praktyce wiadomości nabytych podczas wykładów, w tym podstawowej wiedzy koniecznej do tworzenia i rozumienia oprogramowania, w szczególności dotyczącej podstawowych i zaawansowanych technik obiektowych, na przykładzie języka C++ (analiza, projektowanie i programowanie obiektowe).				
17. Efekty kształcenia:¹				
Nr	Opis efektu kształcenia	Metoda sprawdzenia efektu kształcenia	Forma prowadzenia zajęć	Odniesienie do efektów dla kierunku studiów
1	Ma szczegółową wiedzę w zakresie programowania strukturalnego	SP	WM	K1A_W10
2	Ma szczegółową wiedzę w zakresie programowania obiektowego	EP	WM	K1A_W10
3	Potrafi zaimplementować prosty program strukturalny. Korzysta z instrukcji sterujących przebiegiem programu, obsługuje pliki, obsługuje wskaźniki.	CL	L	K1A_U01, K1A_U24, K1A_U20
4	Potrafi samodzielnie zaprojektować oraz zrealizować prosty program strukturalny	CL	L	K1A_U01, K1A_U24, K1A_U20

¹ należy wskazać ok. 5 – 8 efektów kształcenia

5	Potrafi pisać programy obiektowe. Wie jak zadeklarować klasę, rozumie dziedziczenie, posługuje się strumieniami.	CL	L	K1A_U01, K1A_U24, K1A_U20
6	Potrafi tworzyć programy obiektowo zorientowane	CL	L	K1A_U01, K1A_U24, K1A_U20
7	Umie wykorzystać szablony	CL	L	K1A_U01, K1A_U24, K1A_U20
8	Potrafi samodzielnie zaprojektować oraz zrealizować złożony program zorientowany obiektowo	RP	P	K1A_U01, K1A_U24, K1A_U20

18. Formy zajęć dydaktycznych i ich wymiar (liczba godzin)

W. 25 Ćw. - L. 25 P. 10 Sem. -

19. Treści kształcenia:

Tematyka wykładów:

Wprowadzenie: pierwsze programy przykładowe. Zmienne, typy proste i złożone, operatory i wyrażenia. Instrukcje i sterowanie. Funkcje, struktura programów i klasy pamięci. Tablice, wskaźniki i dynamiczne przydzielanie pamięci. Struktury, unie i dynamiczne struktury danych. Preprocesor, pliki nagłówkowe i biblioteki. Techniki obiektowe, na przykładzie języka C++. Plan wykładu: Paradygmat programowania obiektowego. Zarys programowania obiektowego w C++. Klasy, obiekty i metody (funkcje składowe). Funkcje i klasy zaprzyjaźnione. Konstruktory i destruktory. Cykl życia obiektów. Przeciążone operatory i konwersje definiowane przez użytkownika. Dziedziczenie. Polimorfizm. Klasy abstrakcyjne. Dziedziczenie wielobazowe i klasy wirtualne. Strumienie. Wyjątki. Wzorce. Informacja o typach czasu wykonania (RTTI). Elementy biblioteki standardowej C++ STL.

Tematyka laboratorium:

Typy danych, operatory, instrukcje. Typy tablicowe i wskaźnikowe. Biblioteki i funkcje biblioteczne. Praca z plikami. Struktury i unie. Możliwości C++ w części nie obiektowej, Klasy, Konstruktory i destruktory, Operatory przeciążone, Dziedziczenie, Dziedziczenie wielobazowe, Polimorfizm, RTTI, Strumienie, Wyjątki, Wzorce.

20. Egzamin: tak

21. Literatura podstawowa:

- B. W. Kernigan, D.M.Ritche, Język ANSI C, WNT, Warszawa 1994 (lub późniejsze)
- B. W. Kernighan, D.M.Ritche, Ćwiczenia z języka C, WNT, Warszawa 1994 (lub późniejsze)
- B. Stroustrup, Język C++, W-wa, WNT, 1994 (lub późniejsze)
- International Standard for Information Systems—Programming Language C++, ANSI

22. Literatura uzupełniająca:

- Jerzy Kisilewicz, Język C w środowisku Borland C++ - Wrocław : Oficyna Wydawnicza Politechniki Wrocławskiej , 1998
- Jerzy Kisilewicz, Język C++ : programowanie obiektowe - Wrocław : Oficyna Wydawnicza Politechniki Wrocławskiej, 1999
- A. Zalewski, Programowanie w językach C i C++ z wykorzystaniem pakietu Borland C++. Wyd. Nakom, Poznań, 1997
- J. Grębosz, Symfonia C++ Standard, tom I/II, Edition2000, Kraków, 2006
- J. Grębosz, Pasja C++, tom I/II. Edition2000, Kraków
- Nicolai M. Josuttis: C++ Biblioteka standardowa Podręcznik Programisty, Helion 2003
- C. Delannoy, Ćwiczenia z języka C++, W-wa, WNT, 1994 (lub późniejsze)
- B. Stroustrup, Projektowanie i rozwój języka C++.

23. Nakład pracy studenta potrzebny do osiągnięcia efektów kształcenia

Lp.	Forma zajęć	Liczba godzin kontaktowych / pracy studenta
1	Wykład	25 / 50
2	Ćwiczenia	- / -
3	Laboratorium	25 / 90
4	Projekt	10 / 100
5	Seminarium	- / -
6	Inne (przygotowanie do egzaminu)	- / 30
	Suma godzin	60 / 270

24. Suma wszystkich godzin: 330**25. Liczba punktów ECTS:² 6 (sem.II), 5 (sem.III)****26. Liczba punktów ECTS uzyskanych na zajęciach z bezpośrednim udziałem nauczyciela akademickiego: 2****27. Liczba punktów ECTS uzyskanych na zajęciach o charakterze praktycznym (laboratoria, projekty): 2****26. Uwagi: -**

Zatwierdzono:

.....
(data i podpis prowadzącego).....
(data i podpis dyrektora instytutu/kierownika katedry/
Dyrektora Kolegium Języków Obcych/kierownika lub
dyrektora jednostki międzywydziałowej)

² 1 punkt ECTS – 30 godzin.