

(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: 2014/2015				
4. Forma kształcenia: studia drugiego stopnia				
5. Forma studiów: studia niestacjonarne				
6. Kierunek studiów: INFORMATYKA WYDZIAŁ AEII				
7. Profil studiów: ogólnoakademicki				
8. Specjalność: WSZYSTKIE				
9. Semestr: 1, 2				
10. Jednostka prowadząca przedmiot: Instytut Informatyki / Zakład Oprogramowania RAu2				
11. Prowadzący przedmiot: dr inż. Dariusz Myszor				
12. Przynależność do grupy przedmiotów: przedmioty wspólne				
13. Status przedmiotu: obowiązkowy				
14. Język prowadzenia zajęć: polski				
15. Przedmioty wprowadzające oraz wymagania wstępne: brak				
16. Cel przedmiotu: Przedmiot ma przekazać wiedzę niezbędną do zrozumienia, projektowania i pisania programów oraz umiejętność implementacji oprogramowania na przykładzie języków C, 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++ i 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
W1	Ma szczegółową wiedzę w zakresie programowania strukturalnego.	Egzamin	Wykład / Ćwiczenia	K1A_W10
W2	Ma szczegółową wiedzę w zakresie programowania obiektowego.	Egzamin	Wykład / Ćwiczenia	K1A_W10
W3	Potrafi zaimplementować prosty program strukturalny. Korzysta z instrukcji sterujących przebiegiem programu, obsługuje pliki, obsługuje wskaźniki.	Kolokwium	Wykład / Ćwiczenia	K1A_U01, K1A_U24, K1A_U20
W4	Potrafi samodzielnie zaprojektować oraz zrealizować prosty program strukturalny	Kolokwium	Wykład / Ćwiczenia	K1A_U01, K1A_U24, K1A_U20

¹ należy wskazać ok. 5 – 8 efektów kształcenia

W5	Potrafi pisać programy obiektowe. Wie jak zadeklarować klasę, rozumie dziedziczenie, posługuje się strumieniami.	Kolokwium	Wykład / Ćwiczenia	K1A_U01, K1A_U24, K1A_U20
W6	Potrafi tworzyć programy zorientowane obiektowo	Kolokwium	Wykład / Ćwiczenia	K1A_U01, K1A_U24, K1A_U20
W7	Potrafi samodzielnie zaprojektować oraz zrealizować złożony program zorientowany obiektowo	Projekt	Ćwiczenia / projekt	K1A_U01, K1A_U24, K1A_U20
W8	Potrafi analizować algorytmy oceniać ich złożoność obliczeniową i oszacować złożoność problemów	Projekt	Ćwiczenia / projekt	K1A_U01, K1A_U24, K1A_U20
W9	Rozumie potrzebę uczenia się przez całe życie; potrafi inspirować i organizować proces uczenia się innych osób	Projekt	Ćwiczenia / projekt	K1A_U01, K1A_U24, K1A_U20
18. Formy zajęć dydaktycznych i ich wymiar (liczba godzin)				
W. 15 Ćw. 0 L. 15 P. 0 Sem. 1				
W. 15 Ćw. 0 L. 15 P. 15 Sem. 2				
19. Treści kształcenia:				
Tematyka wykładów: 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++ i języka C#. Paradygmat programowania obiektowego. Zarys programowania obiektowego w C++ i 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. Dostęp do baz danych w C# i Linq. Opis platformy .Net. 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. Podstawowe elementy języka C#, tworzenie klas, obsługa bazy danych, technologii WebService, tworzenie UI, zastosowanie delegatów. (oddzielnie dla każdej z form zajęć dydaktycznych W./Ćw./L./P./Sem.)				
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)
C#. Programowanie. Wydanie VI Ian Griffiths, Matthew Adams, Jesse Liberty

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

23. Nakład pracy studenta potrzebny do osiągnięcia efektów kształcenia

Lp.	Forma zajęć	Liczba godzin kontaktowych / pracy studenta
1	Wykład	30h / 30h
2	Ćwiczenia	0h / 0h
3	Laboratorium	30h / 75h
4	Projekt	15h / 120h
5	Seminarium	0/0
6	Inne	0/0
	Suma godzin	75h / 225h

24. Suma wszystkich godzin: 75 + 225 = 300**25. Liczba punktów ECTS: 4 (sem1), 6 (sem2)****26. Liczba punktów ECTS uzyskanych na zajęciach z bezpośrednim udziałem nauczyciela akademickiego 6****27. Liczba punktów ECTS uzyskanych na zajęciach o charakterze praktycznym (laboratoria, projekty) 4****26. Uwagi: -**

Zatwierdzono:

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