

KARTA PRZEDMIOTU

Kod przedmiotu	0533.6.SDM1.B/C.PP	
Nazwa przedmiotu w języku	polskim	<i>Podstawy programowania</i> <i>Basics of programming</i>
	angielskim	

1. USYTUOWANIE PRZEDMIOTU W SYSTEMIE STUDIÓW

1.1. Kierunek studiów	Systemy diagnostyczne w medycynie
1.2. Forma studiów	Stacjonarne
1.3. Poziom studiów	Pierwszego stopnia
1.4. Profil studiów*	Ogólnoakademicki
1.5. Osoba przygotowująca kartę przedmiotu	dr hab. Grzegorz Stefanek, prof. UJK
1.6. Kontakt	grzegorz.stefanek@ujk.edu.pl

2. OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

2.1. Język wykładowy	polski
2.2. Wymagania wstępne*	brak

3. SZCZEGÓŁOWA CHARAKTERYSTYKA PRZEDMIOTU

3.1. Forma zajęć	Wykład: 15h, Laboratorium: 30 h	
3.2. Miejsce realizacji zajęć	Zajęcia dydaktyczne w pomieszczeniach UJK	
3.3. Forma zaliczenia zajęć	Zaliczenie z oceną	
3.4. Metody dydaktyczne	Wykład – wykład informacyjny, wykład problemowy, ćwiczenia laboratoryjne	
3.5. Wykaz literatury	podstawowa	Piotr Wróblewski, „Algorytmy, struktury danych i techniki programowania”, Wydanie VI, Helion 2019 B.W. Kernighan, D.M. Ritchie "Język ANSI C. Programowanie,", Helion, 2020
	uzupełniająca	Maciej. M. Sysło „Algorytmy”, Helion, 2016 Alfred V.Abo, John E.Hopcroft, Jefferey D.Ullman „Algorytmy i struktury danych = programy”, Helion, 2003 Stephen Prata, „Język C++. Szkoła programowania”, Helion, 2012

4. CELE, TREŚCI I EFEKTY UCZENIA SIĘ

<p>4.1. Cele przedmiotu (z uwzględnieniem formy zajęć)</p> <p>Wykład</p> <p>C1. Przedstawienie podstawowych pojęć programistycznych oraz omówienie cyklu przygotowania oprogramowania.</p> <p>C2. Omówienie podstawowych elementów języka C/C++ wraz z przykładami zastosowania.</p> <p>Laboratorium</p> <p>C1. Nabycie umiejętności budowania schematów blokowych jako formy zapisu algorytmów dla zadań informatycznych.</p> <p>C2. Kształtowanie umiejętności implementacji prostych zadań informatycznych w formie kodów w C/C++.</p>
<p>4.2. Treści programowe (z uwzględnieniem formy zajęć)</p> <p>Wykład</p> <p>Podstawowe pojęcia z zakresu programowania: algorytm, schemat blokowy, język programowania, cykl programowania, kod źródłowy i wynikowy, kompilacja i interpretacja, programowanie strukturalne i obiektowe, środowisko programistyczne</p> <p>Schematy blokowe liniowe, warunkowe i cykliczne. Algorytmy iteracyjne i rekurencyjne.</p> <p>Podstawowe struktury danych: tablice, kolejki, stosy, drzewa i grafy.</p> <p>Podstawy programowania w języku C/C++.</p> <p>Identyfikatory, stałe i zmienne. Typy zmiennych. Operatory arytmetyczne, logiczne, bitowe i operatory porównania.</p> <p>Priorytet operatorów i kolejność wykonywania działań.</p> <p>Instrukcje warunkowe i iteracyjne. Funkcje i procedury. Funkcje wejścia/wyjścia. Obsługa plików.</p> <p>Wskaźniki i referencje. Operowanie wskaźnikami. Klasy i obiekty. Dziedziczenie, polimorfizm i metody wirtualne.</p> <p>Laboratorium (praktyczne realizacje schematów blokowych i ich implementacje w języku C/C++)</p> <p>Proste schematy/programy warunkowe: rozwiązywanie równań liniowych i kwadratowych, porównywanie elementów zbioru</p> <p>Proste schematy/programy numeryczne: konwersja liczb między systemami liczenia, wyznaczanie największego</p>

wspólnego dzielnika liczb (metoda Euklidesa), sumowanie szeregów dla matematycznej liczby π i liczby Eulera
 Proste programy iteracyjne: znajdowanie najmniejszego, największego elementu zbioru, znajdowanie średniej i wariancji zestawu liczb
 Programy przeszukiwania i sortowania: przeszukiwanie liniowe, binarne, interpolacyjne; sortowanie przez wybór, sortowanie bąbelkowe i pozycyjne
 Programy wykorzystujące metodę dziel i zwyciężaj: sortowanie przez scalanie, znajdowanie miejsca zerowego funkcji
 Wykorzystanie tablic wielowymiarowych: znajdowanie śladu tablicy/macierzy, sumowanie i mnożenie tablic
 Programy rekurencyjne: liczenie silni, znajdowanie wartości wielomianu metodą Hornera, wyznaczanie wartości liczb Fibonacciego
 Niedeterministyczne metody informatyczne: metoda Monte-Carlo
 Budowanie prostych klas/obiektów i metod tych klas.

4.3. Przedmiotowe efekty uczenia się

Efekt	Student, który zaliczył przedmiot	Odniesienie do kierunkowych efektów uczenia się
w zakresie WIEDZY:		
W01	Objaśnia podstawowe algorytmy.	SDM1A_W06
W02	Zna podstawowe typy zmiennych, struktury danych oraz operacje we/wy.	SDM1A_W06
W03	Zna operatory, instrukcje i inne elementy języka C.	SDM1A_W06
w zakresie UMIEJĘTNOŚCI:		
U01	Opracowuje program na podstawie algorytmu.	SDM1A_U07
U02	Projektuje aplikacje na zadany temat.	SDM1A_U07
w zakresie KOMPETENCJI SPOŁECZNYCH:		
K01	Wykazuje kreatywność w rozwiązywaniu powierzonych problemów algorytmicznych i programistycznych oraz rozumie potrzebę poszukiwania optymalnego rozwiązania.	SDM1A_K02 SDM1A_K03
K02	Jest gotów do samodzielnej i odpowiedzialnej pracy w zakresie przygotowania prostych aplikacji programistycznych.	SDM1A_K02 SDM1A_K03

4.4. Sposoby weryfikacji osiągnięcia przedmiotowych efektów uczenia się

Efekty przedmiotowe (symbol)	Sposób weryfikacji (+/-)																				
	Egzamin ustny/pisemny*			Kolokwium*			Projekt*			Aktywność na zajęciach*			Praca własna*			Praca w grupie*			Inne (jakie?)* np. test - stosowany w e-learningu		
	Forma zajęć			Forma zajęć			Forma zajęć			Forma zajęć			Forma zajęć			Forma zajęć			Forma zajęć		
	W	C	L	W	C	L	W	C	L	W	C	L	W	C	L	W	C	L	W	C	L
W01				X		X						X			X						
W02				X		X						X			X						
W03				X		X						X			X						
U01				X		X						X			X						
U02				X		X						X			X						
K01				X		X						X			X						
K02															X						

*niepotrzebne usunąć

4.5. Kryteria oceny stopnia osiągnięcia efektów uczenia się

Forma zajęć	Ocena	Kryterium oceny
wykład (W) (w tym e-learning)	3	co najmniej 50% i nie więcej niż 60% łącznej liczby punktów możliwych do uzyskania
	3,5	ponad 60% i nie więcej niż 70% łącznej liczby punktów możliwych do uzyskania
	4	ponad 70% i nie więcej niż 80% łącznej liczby punktów możliwych do uzyskania
	4,5	ponad 80% i nie więcej niż 90% łącznej liczby punktów możliwych do uzyskania
	5	ponad 90% liczby punktów możliwych do uzyskania

laboratorium (L)*	3	co najmniej 50% i nie więcej niż 60% łącznej liczby punktów możliwych do uzyskania
	3,5	ponad 60% i nie więcej niż 70% łącznej liczby punktów możliwych do uzyskania
	4	ponad 70% i nie więcej niż 80% łącznej liczby punktów możliwych do uzyskania
	4,5	ponad 80% i nie więcej niż 90% łącznej liczby punktów możliwych do uzyskania
	5	ponad 90% liczby punktów możliwych do uzyskania

5. BILANS PUNKTÓW ECTS – NAKŁAD PRACY STUDENTA

Kategoria	Obciążenie studenta	
	Studia stacjonarne	Studia niestacjonarne
<i>LICZBA GODZIN REALIZOWANYCH PRZY BEZPOŚREDNIM UDZIALE NAUCZYCIELA /GODZINY KONTAKTOWE/</i>	45	
<i>Udział w wykładach*</i>	15	
<i>Udział w ćwiczeniach, konwersatoriach, laboratoriach*</i>	28	
<i>Udział w egzaminie /kolokwium zaliczeniowym*</i>	2	
<i>SAMODZIELNA PRACA STUDENTA /GODZINY NIEKONTAKTOWE/</i>	30	
<i>Przygotowanie do wykładu*</i>	10	
<i>Przygotowanie do ćwiczeń, konwersatorium, laboratorium*</i>	10	
<i>Przygotowanie do egzaminu /kolokwium*</i>	10	
ŁĄCZNA LICZBA GODZIN	75	
PUNKTY ECTS za przedmiot	3	

**niepotrzebne usunąć*

Przyjmuję do realizacji (data i czytelne podpisy osób prowadzących przedmiot w danym roku akademickim)

.....

