

## KARTA KURSU

Nazwa	<b>Podstawy Programowania</b>
Nazwa w j. ang.	<i>Programming basics</i>

Koordynator	dr Roman Czapla	Zespół dydaktyczny
		dr Roman Czapla dr Zdobysław Świerczyński
Punktacja ECTS*	4	

### Opis kursu (cele kształcenia)

Celem przedmiotu jest nauka programowania proceduralnego w oparciu o język C++ (standard: przynajmniej C++11) oraz metodami projektowania i zapisu algorytmów. Po zakończeniu kursu student będzie znał podstawy programowania w języku C++ i umiał je stosować. Przedmiot prowadzony jest w języku polskim.

### Warunki wstępne

Wiedza	Student zna podstawowe pojęcia związane z programowaniem.
Umiejętności	Student posiada umiejętność opisanego prostych algorytmów.
Kursy	Wstępne kursy nie są wymagane.

### Efekty uczenia się

	Efekt uczenia się dla kursu	Odniesienie do efektów kierunkowych
Wiedza	Po zakończeniu kursu student:	
	<b>W01:</b> zna i rozumie podstawowe zasady budowy algorytmu;	K_W03
	<b>W02:</b> zna elementarne konstrukcje i semantykę języka C++ w tym: <ul style="list-style-type: none"> <li>– potrafi podać ogólną klasyfikację struktur danych, wskazuje różnice pomiędzy typem prostym i złożonym oraz zna sposoby reprezentacji danych w pamięci komputera;</li> <li>– rozumie pojęcie wskaźnika i arytmetyki wskaźnikowej;</li> <li>– rozumie model pamięci i zna pojęcie zakresu zmiennych;</li> </ul>	K_W03
	<b>W03:</b> ma wiedzę na temat funkcji (w tym ich zafasowanych elementów) i rekurencji;	K_W03
	<b>W04:</b> wie jak dynamicznie alokować pamięć i jak pracować z plikami.	K_W03

	Efekt uczenia się dla kursu	Odniesienie do efektów kierunkowych
Umiejętności	Po zakończeniu kursu student:	
	<b>U01:</b> potrafi wybierać i stosować struktury danych zdefiniowane w C++, aby rozwiązać dany problem; <b>U02:</b> potrafi projektować i wdrażać programy z wykorzystaniem istniejących bibliotek C++ zgodnie z paradygmatem proceduralnym; <b>U03:</b> umie kompilować i uruchamiać programy w języku C++ oraz śledzić krokowo ich wykonywanie, analizować i poprawiać błędy składniowe i błędy wykonania.	K_U04, K_U03  K_U04, K_U03  K_U04, K_U03

	Efekt uczenia się dla kursu	Odniesienie do efektów kierunkowych
Kompetencje społeczne	Po zakończeniu kursu student:	
	<b>K01:</b> potrafi pozyskiwać informacje z literatury, dokumentacji (w tym zasobów sieci Internet) niezbędnych do poszerzania własnej wiedzy i zdobywania nowych umiejętności z zakresu programowania i projektowania programów. <b>K02:</b> potrafi przekazywać wiedzę informatyczną w sposób zrozumiały dla innych.	K_K02  K_K01

### Studia stacjonarne

		Organizacja										
Forma zajęć	Wykład (W)	Ćwiczenia w grupach										
		A		K		L		S		P		E
Liczba godzin	20					30						

### Studia niestacjonarne

		Organizacja										
Forma zajęć	Wykład (W)	Ćwiczenia w grupach										
		A		K		L		S		P		E
Liczba godzin	15					20						

## Opis metod prowadzenia zajęć

Podczas pracy laboratoryjnej studenci rozwiązują zadania zdefiniowane przez prowadzącego zajęcia. Na ćwiczeniach na bieżąco omawiane (dyskutowane) są zagadnienia poruszane podczas wykładów.

## Formy sprawdzania efektów uczenia się

	E – learning	Gry dydaktyczne	Ćwiczenia w szkole	Zajęcia terenowe	Praca laboratoryjna	Projekt indywidualny	Projekt grupowy	Udział w dyskusji	Referat	Praca pisemna (esej)	Egzamin ustny	Egzamin pisemny	Inne
W01					X	X		X				X	
W02					X	X		X				X	
W03					X	X		X				X	
W04					X	X		X				X	
U01					X	X		X				X	
U02					X	X		X				X	
U03					X	X		X					
K01					X	X		X					
K02					X	X		X					

Kryteria oceny	<p>Osiągnięcie efektów kształcenia podanych powyżej uprawnia studentów do uzyskania oceny nie wyższej niż dostateczna. Ocenę dobrą lub bardzo dobrą może uzyskać student, który:</p> <ul style="list-style-type: none"> <li>– wykazuje się umiejętnością konstruowania algorytmów o większej złożoności oraz ich efektywną realizacją w języku programowania C++;</li> <li>– prawidłowo dobiera poznane struktury danych do rozwiązywanego problemu;</li> <li>– wyjaśnia mechanizm przekazywania parametrów do funkcji (w tym poprzez referencje);</li> <li>– biegle korzysta ze wskaźników oraz referencji, rozumie arytmetykę oraz notację wskaźnikową;</li> <li>– potrafi pisać zaawansowane programy korzystające z plików tekstowych (zapis/odczyt).</li> </ul>
----------------	--

Uwagi	
-------	--

## Treści merytoryczne (wykaz tematów)

1. Wprowadzenie do algorytmiki i programowania - pojęcie algorytmu i programu komputerowego, sposoby zapisu i reprezentacji algorytmów, charakterystyka języków programowania (w tym wysokiego poziomu), proces kompilacji, sposoby reprezentacji liczb w systemie komputerowym.
2. Język C++. Podstawowe typy danych, deklaracje – zmienne i stałe, podstawowe (wybrane) operatory i wyrażenia, priorytety operatorów.
3. Struktura programu w języku C++, komentarze, formatowanie kodu źródłowego, wybrane dyrektywy preprocesora.

4. Obsługa standardowego wejścia/wyjścia. Kompilowanie i śledzenie wykonania programu.
5. Składnia i semantyka instrukcji - instrukcje przypisania, grupujące, sterujące - warunkowe iteracyjne (pętle i pętle zakresowe). Instrukcja switch, break i continue.
6. Funkcje – prototyp, definiowanie, wywołanie, parametry, argumenty i zmienne lokalne funkcji, mechanizm przekazywania parametrów. Parametry referencyjne i domyślne.
7. Model pamięci i przestrzenie nazw.
8. Tablice jednowymiarowe, tablice dwuwymiarowe, wektory. Funkcje korzystające z tablic i wektorów.
9. Wskaźniki i referencje w języku C++.
10. Funkcje rekurencyjne. Funkcje lambda i inline. Przeciążenie funkcji.
11. Struktury - wskaźniki do struktur, tablice struktur, funkcje a struktury. Ciągi tekstowe (w stylu C) i klasa string. Dynamiczna alokacja pamięci. Tablice i wektory wskaźników.
12. Strumienie. Odczytywanie i zapisywanie plików tekstowych. Argumenty wiersza poleceń.

#### Wykaz literatury podstawowej

Wybrane fragmenty:

1. Cay S. Horstmann, *Wprowadzenie do C++. Efektywne nauczanie. Wydanie III*, Helion, 2021;
2. J. Grębosz, *Opus magnum C++ 11. Programowanie w języku C++. Wydanie II poprawione (komplet)*, Helion, 2020;
3. J. Grębosz, *Opus magnum C++. Misja w nadprzestrzeń C++14/17. Tom 4*, Helion, 2020;
4. B. Stroustrup, *Język C++. Kompendium wiedzy. Wydanie IV*, Helion 2014;
5. M. Matlak, *Wprowadzenie do C++*, Helion, 2022;
6. B. Cyganek, *Programowanie w języku C++*, PWN, 2023.

#### Wykaz literatury uzupełniającej

1. T. Jaśniewski, *C++ Zbiór zadań z rozwiązaniami*, Helion, 2023;
2. J. Lospinoso, *C++ Intensywny kurs*, PWN, 2021;
3. B. Stroustrup, *Programowanie. Teoria i praktyka z wykorzystaniem C++*. Wydanie III, Helion 2020
4. B. Stroustrup, *C++. Podróż po języku dla zaawansowanych*. Wydanie III, Helion, 2023;
5. I. Čukić, *Programowanie funkcyjne w języku C++*. Tworzenie lepszych aplikacji, Helion 2019;
6. S. Prata, *Język C++. Szkoła programowania*. Wydanie VI, Helion, 2012;
7. D. Farbaniec, *C++20. Laboratorium*, Helion, 2022;
8. M. Bancila, *Nowoczesny C++*. Zbiór praktycznych zadań dla przyszłych ekspertów, Helion, 2019.

Bilans godzinowy zgodny z CNPS (Całkowity Nakład Pracy Studenta) – **studia stacjonarne**

liczba godzin w kontakcie z prowadzącymi	Wykład	20
	Konwersatorium (ćwiczenia, laboratorium itd.)	30
	Pozostałe godziny kontaktu studenta z prowadzącym	5
liczba godzin pracy studenta bez kontaktu z prowadzącymi	Lektura w ramach przygotowania do zajęć	15
	Przygotowanie krótkiej pracy pisemnej lub referatu po zapoznaniu się z niezbędną literaturą przedmiotu	0
	Przygotowanie projektu lub prezentacji na podany temat (praca w grupie)	
	Przygotowanie do egzaminu/zaliczenia	30
Ogółem bilans czasu pracy		100
Liczba punktów ECTS w zależności od przyjętego przelicznika		4

Bilans godzinowy zgodny z CNPS (Całkowity Nakład Pracy Studenta) – **studia niestacjonarne**

liczba godzin w kontakcie z prowadzącymi	Wykład	15
	Konwersatorium (ćwiczenia, laboratorium itd.)	20
	Pozostałe godziny kontaktu studenta z prowadzącym	5
liczba godzin pracy studenta bez kontaktu z prowadzącymi	Lektura w ramach przygotowania do zajęć	15
	Przygotowanie krótkiej pracy pisemnej lub referatu po zapoznaniu się z niezbędną literaturą przedmiotu	0
	Przygotowanie projektu lub prezentacji na podany temat (praca w grupie)	
	Przygotowanie do egzaminu/zaliczenia	30
Ogółem bilans czasu pracy		100
Liczba punktów ECTS w zależności od przyjętego przelicznika		4