



## KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Wprowadzenie do programowania [S1Cybez1>WdP]

### Przedmiot

Kierunek studiów

Cyberbezpieczeństwo

Rok/Semestr

1/1

Studia w zakresie (specjalność)

–

Profil studiów

ogólnoakademicki

Poziom studiów

pierwszego stopnia

Język oferowanego przedmiotu

polski

Forma studiów

stacjonarne

Wymagalność

obligatoryjny

### Liczba godzin

Wykład

30

Laboratorium

30

Inne

0

Ćwiczenia

0

Projekty/seminaria

0

### Liczba punktów ECTS

5,00

### Koordynatorzy

prof. dr hab. inż. Jerzy Nawrocki  
jerzy.nawrocki@put.poznan.pl

### Wykładowcy

### Wymagania wstępne

Wymagana jest wiedza z matematyki i informatyki określona w podstawie programowej dla liceum ogólnokształcącego i technikum - poziom podstawowy.

### Cel przedmiotu

Celem przedmiotu jest przekazanie studentom zaawansowanej wiedzy na temat różnych paradygmatów programowanie i najważniejszych obszarów inżynierii oprogramowania oraz wsparcie ich w rozwijaniu umiejętności w tym zakresie.

### Przedmiotowe efekty uczenia się

Wiedza:

Przedmiot daje wiedzę na temat różnych paradygmatów programowania i związanych z nimi języków oraz najważniejszych obszarów inżynierii oprogramowania (K1\_W06).

Umiejętności:

Kończąc przedmiot student ma podstawowe umiejętności w zakresie programowania z wykorzystaniem różnych paradygmatów i języków programowania (K1\_U10) oraz oceny jakości oprogramowania

(K1\_U09).

Kompetencje społeczne:

Przedmiot przyczynia się do podnoszenia świadomości znaczenia pracy własnej i zespołowej (K1\_K05).

### Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

1. Krótkie quizy przeprowadzane w trakcie wykładu za pomocą platformy eKursy (Moodle)
2. Ocena bieżąca w ramach ćwiczeń laboratoryjnych
3. Test indywidualny
4. Zawody zespołowe
5. Egzamin

Zasady zaliczania przedmiotu i dokładne progi zaliczeniowe zostaną przekazane studentom na początku semestru z wykorzystaniem uczelnianych systemów elektronicznych oraz na pierwszych zajęciach (w każdej formie zajęć).

### Treści programowe

Zakres tematyczny przedmiotu obejmuje języki programowania, C/C++ i Python, podstawy asemblera NASM, wyrażenia regularne, przetwarzanie regułowe, złożoność obliczeniową, język SQL i jego łączenie z językami C/C++ i Python, przetwarzanie równoległe w językach C/C++ i Python, język JavaScript, programowanie mikrokontrolerów i Arduino, przypadki użycia, testowanie i przeglądy kodu, wybrane diagramy języka UML, wybrane wzorce projektowe.

### Tematyka zajęć

Programowanie imperatywne  
Język asemblera  
Funkcje i procedury  
Liczby rzeczywiste i metody numeryczne  
Programowanie obiektowe  
Przetwarzanie tekstu i wyrażenia regularne  
Programowanie regułowe  
Złożoność obliczeniowa  
Bazy danych i SQL  
Przetwarzanie współbieżne  
Przetwarzanie sterowane zdarzeniami i JavaScript  
Programowanie systemów wbudowanych  
Specyfikacja wymagań  
Ocena jakości oprogramowania  
Architektura i projektowanie oprogramowania

### Metody dydaktyczne

Wykład z materiałami wykładowymi dostępnymi na platformie eKursy  
Powtórki materiału z poprzedniego wykładu w formie quizów realizowanych na platformie eKursy  
Rozwiązywanie zadań programistycznych w ramach ćwiczeń laboratoryjnych  
Zawody zespołowe jako praktyczne wprowadzenie do pracy zespołowej

### Literatura

Podstawowa:

Kernighan, Brian W., and Dennis M. Ritchie. "Język ANSI C: programowanie", Helion, 2020.

Lutz, Mark. "Python. Wprowadzenie", O'Reilly, 2022.

Uzupełniająca:

Nawrocki, Jerzy R. "Programowanie komputerów IBM PC w języku asemblera metodą systematyczną", Wydawnictwo Politechniki Poznańskiej, 1991.

### Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	130	5,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	60	2,50
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwίων/egzaminu, wykonanie projektu)	70	2,50