Państwowa Wyższa Szkoła Zawodowa
im. H. Cegielskiego w Gnieźnie

**Instytut Informatyki i Telekomunikacji**

| Nazwa modułu/przedmiotu | Kod |
|---|---|
| **Wprowadzenie do programowania** | |

# KARTA OPISU MODUŁU KSZTAŁCENIA

| Kierunek studiów | Profil kształcenia | Kurs (obligatoryjny/obieralny) |
|---|---|---|
| **Informatyka** | **praktyczny** | **obligatoryjny** |
| Specjalność: **Wszystkie specjalności** | Przedmiot oferowany w języku: polskim | Punkty ECTS (liczba i %): **5** |
| Stopień studiów: 1 | Obszar(y) kształcenia: **nauki techniczne** | 100% |
| Status przedmiotu w programie studiów | | |
| (podstawowy, kierunkowy, inny: **inny** | ogólnouczelniany, z innego kierunku: ogólnouczelniany | |

Forma studiów i godziny zajęć w danym semestrze

| **stacjonarne** | | | | | **niestacjonarne** | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Wykłady | Ćwiczenia | Laborat. | Projekty / seminaria | Rok/ Semestr | Wykłady | Ćwiczenia | Laborat. | Projekty / seminaria | Rok/ Semestr |
| 30 | - | 30 | | **1/1** | **16** | **-** | **16** | **-** | **-** |

Jednostka prowadząca przedmiot: **Instytut Informatyki i Telekomunikacji**


**Osoba odpowiedzialna za przedmiot / wykładowca:**

Dr inż. Piotr Remlein
e-mail: piotr.remlein@put.poznan.pl
tel. 61 424 2942
Instytut Informatyki i Telekomunikacji
ul. Ks. S. Wyszyńskiego 36, 62-200 Gniezno

**Lista osób prowadzących zajęcia:**

Dr inż. Piotr Remlein
e-mail: piotr.remlein@put.poznan.pl
tel. 61 424 2942
Instytut Informatyki i Telekomunikacji
ul. Ks. S. Wyszyńskiego 36, 62-200 Gniezno


## Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych:

| | | |
|---|---|---|
| 1 | **Wiedza:** | Podstawowe wiadomości z zakresu podstaw logiki i matematyki |
| 2 | **Umiejętności:** | Umiejętność efektywnego samokształcenia w dziedzinach związanych z informatyką jako wybranym kierunkiem studiów |
| 3 | **Kompetencje społeczne** | Ma świadomość konieczności poszerzania swoich kompetencji oraz gotowość do podjęcia współpracy w ramach zespołu |

## Cel przedmiotu:

Przedstawienie podstaw architektury komputerów I sposobów reprezentowania danych w pamięci komputera. Poznanie zasad zapisu algorytmów za pomocą schematów blokowych, a następnie za pomocą języka programowania C. Nabycie praktycznej umiejętności przygotowania, uruchamiania i testowania programów. Przedmiot zapoznaje z podstawami programowania strukturalnego niezbędnymi dla każdego informatyka.

## Efekty kształcenia

| **Wiedza.** W wyniku przeprowadzonych zajęć student: | | Odniesienie do Kierunkowych Efektów Kształcenia |
|---|---|---|
| 01 | Ma podstawową wiedzę w zakresie architektury komputerów i działania jego podstawowych elementów składowych. Potrafi opisać cykl pracy procesora i umie scharakteryzować język wewnętrzny procesora. Zna systemy zapisu liczb, wyznaczania reprezentacji liczb całkowitych i rzeczywistych oraz wykonywania podstawowych operacji arytmetycznych na tych reprezentacjach | **K_W06 ++** |
| 02 | Ma uporządkowaną i podbudowaną teoretycznie wiedzę w zakresie zasad zapisu algorytmów za pomocą schematów blokowych oraz zna rodzaje i zasady programowania, w tym za pomocą języka programowania wyższego poziomu | **K_W07 +++** |

| | | |
|---|---|---|
| 03 | Ma szczegółową wiedzę na temat języka programowania C (strukturalnej części języka programowania Zna sposoby definiowania danych i struktur danych, modularyzacji oprogramowania, korzystania z plików dyskowych. | **K_W07 +++** |
| **Umiejętności.** W wyniku przeprowadzonych zajęć student będzie potrafił: | | Odniesienie do Kierunkowych Efektów Kształcenia |
| 01 | Stosować wiedzę z zakresu podstaw programowania do formułowania prostych algorytmów iteracyjnych i rekurencyjnych oraz zapisywania ich w języku programowania C | **K_U15 +++** |
| 02 | Stosować wiedzę z zakresu podstaw programowania do definiowania struktur danych oraz struktur informacyjnych przechowywanych w plikach dyskowych i przetwarzania tych struktur za pomocą języka C | **K_U15 ++**<br>**K_U16 ++** |
| 03 | Stosować wiedzę z zakresu technik programowania do uruchamiania i testowania programów (w tym interakcyjnych) składających się z wielu modułów. | **K_U15 ++** |
| **Kompetencje społeczne.** W wyniku przeprowadzonych zajęć student zdobędzie następujące kompetencje: | | Odniesienie do Kierunkowych Efektów Kształcenia |
| 01 | Rozumie potrzebę permanentnego kształcenia się i przekazywania w sposób zrozumiały informacji z najbliższym otoczeniem w działalności zawodowej. | **K_K01 +** |
| 02 | Rozumie pozatechniczne (w tym ekologiczne) skutki swojego działania i jego wpływu na środowisko, szczególnie w zakresie odpowiedzialności za wytworzony produkt programistyczny | **K_K02 +**<br>**K_K04 +** |

**Sposoby sprawdzenia efektów kształcenia**

Wykład

- ocenianie ciągłe na każdych zajęciach (premiowanie aktywności i jakości percepcji).

Ćwiczenia:

- sprawdziany weryfikujące przyrost wiedzy z zakresu architektury komputerów, zapisu danych w pamięci komputera, arytmetyki binarnej, języka programowania C,
- sprawdziany weryfikujące przyrost umiejętności tworzenia interakcyjnych programów iteracyjnych i rekurencyjnych, korzystających z plików dyskowych, zapisywanych za pomocą języka C
- ocenianie ciągłe, na każdych zajęciach - premiowanie przyrostu umiejętności posługiwania się poznanymi zasadami i metodami;

Uzyskiwanie punktów dodatkowych za aktywność podczas zajęć, a szczególnie za:

- samodzielne rozszerzenie zakresu wiedzy dotyczącej podstaw programowania
- efektywność zastosowania zdobytej wiedzy podczas rozwiązywania zadanego problemu;
- uwagi związane z udoskonaleniem materiałów dydaktycznych;
- wskazywanie trudności percepcyjnych umożliwiające bieżące doskonalenia procesu dydaktycznego.

**Treści programowe**

Definicja komputera. Algorytm, program, dane i struktury danych. Język schematów blokowych. Przykładowe algorytmy. Schemat cyklu rozkazowego procesora. Architektura prostego, przykładowego procesora: rejestry, lista rozkazów. Przykładowe programy zapisane w języku wewnętrznym. Arytmetyka binarna. Język C : strukturalna część proceduralnego języka programowania C++. Alfabet języka, identyfikatory, typy danych liczbowych, zmienne typów numerycznych. Wskaźniki, zmienne wskaźnikowe, operator wyznaczania wskaźnika, operator dostępu pośredniego. Podstawowe funkcje wprowadzania i wyprowadzania danych. Typ logiczny, typy wyliczeniowe. Wyrażenia arytmetyczne i logiczne: operatory arytmetyczne, logiczne, bitowe. Instrukcje proste i złożone. Instrukcje warunkowe, instrukcja przełącznika. Instrukcje tworzenia pętli. Struktury danych: tablice jedno i wielowymiarowe, struktury, unie. Modularyzacja programu: definiowanie i wywoływanie funkcji. Funkcje rekurencyjne. Wskaźniki funkcji, tablice wskaźników funkcji i ich wykorzystanie. Przeciążanie identyfikatorów funkcji. Funkcje biblioteczne umożliwiające korzystanie z plików dyskowych.

**Literatura podstawowa:**

1. J. Kniat, Wprowadzenie do programowania, Wydawnictwo PWSZ w Gnieźnie, 2013
2. J. Kniat, Programowanie w języku C++, Wydawnictwo NAKOM, Poznań 2003. (wersja elektroniczna dostępna w Wielkopolskiej Bibliotece Cyfrowej : www.wbc.poznan.pl)

B.W. Kernighan, D.M. Ritchie, Język C, Wydawnictwa Naukowo – Techniczne, 2007

**Literatura uzupełniająca:**

1. J. Gręboszz, Symfonia C++, Wydawnictwo Edition 2000, Warszawa 2006
2. M. J. Kubiak, C++. Zadania z programowania z przykładowymi rozwiązaniami, Helion, 2011
3. M. Tłuczek, Programowanie w języku C. Ćwiczenia praktyczne. Helion, 2011

Obciążenie pracą studenta

| Studia | stacjonarne | | niestacjonarne | |
|---|---|---|---|---|
| forma aktywności | godziny | ECTS | godziny | ECTS |
| Łączny nakład pracy [1)] | 120 | 5 | 120 | 5 |
| Zajęcia wymagające indywidualnego kontaktu z nauczycielem [2)] | 70 | 3 | 50 | 2 |
| Zajęcia o charakterze praktycznym [3)] | 60 | 3 | 60 | 3 |
| Praca własna studenta [4)] | 50 | 2 | 70 | 3 |

Uwagi

1) łączne obciążenie studenta: G – sumaryczna liczba godzin oraz s – suma pkt. ECTS jest równa dla st. stacjonarnych i niestacjonarnych;

2) zajęcia dydaktyczne {w+c+L+p} + konsultacje +egzamin:
dla stacjonarnych liczba godzin > 50 % godzin z poz1.,
dla niestacjonarnych liczba godzin < 50% z poz.1).;

3) Zajęcia laboratoryjne+przygotowanie do tych zajęć+opracowanie sprawozdań+zajęcia projektowe+przygotowanie do zajęć projektowych+konsulatcje w sprawie projektów+realizacja projektu;

4) Pozycje 2. i 4. dają w sumie liczbę godzin i pkt ECTS podaną w pozycji 1.