**Nazwa przedmiotu:**

Programming of Photonics Devices

**Koordynator przedmiotu:**

dr inż. Sławomir Paśko

**Status przedmiotu:**

Obowiązkowy

**Poziom kształcenia:**

Studia I stopnia

**Program:**

Mechatronics

**Grupa przedmiotów:**

Obowiązkowe

**Kod przedmiotu:**

PPD

**Semestr nominalny:**

6 / rok ak. 2020/2021

**Liczba punktów ECTS:**

4

**Liczba godzin pracy studenta związanych z osiągnięciem efektów uczenia się:**

1) Liczba godzin, które wymagają obecności nauczyciela - 45, w tym
a) obecność na wykładach - 15 godzin;
b) obecność na ćwiczeniach - 15 godzin;
c) spotkania konsultacyjne - 15 godzin;
2) Liczba godzin samodzielnej pracy studenta - 55 godzin, w tym:
a) przygotowanie do testu teoretycznego - 10 godzin;
b) przygotowanie do testu praktycznego - 10 godzin;
c) opracowanie programu w ramach projektu - 35 godzin;
Suma godzin: 100 godzin = 4ECTS

**Liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich:**

1) Liczba godzin, w których wymagana jest obecność nauczyciela - 45 godzin, w tym
a) obecność na wykładach - 15 godzin;
b) obecność na ćwiczeniach - 15 godzin;
c) spotkania konsultacyjne - 15 godzin;
Suma godzin: 55 godzin = 2 ECTS

**Język prowadzenia zajęć:**

angielski

**Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym:**

a) obecność na ćwiczeniach - 15 godzin;
b) przygotowanie do testu praktycznego - 10 godzin;
c) opracowanie programu w ramach projektu - 35 godzin;
Suma godzin: 70 godzin = 2 ECTS

**Formy zajęć i ich wymiar w semestrze:**

|  |  |
| --- | --- |
| Wykład: | 15h |
| Ćwiczenia: | 15h |
| Laboratorium: | 0h |
| Projekt: | 15h |
| Lekcje komputerowe: | 0h |

**Wymagania wstępne:**

Kurs inżynierski z matematyki. Podstawy programowania strukturalnego.

**Limit liczby studentów:**

60

**Cel przedmiotu:**

Znajomość podstaw programowania obiektowego na przykładzie C ++. Podstawowa znajomość języka UML.

**Treści kształcenia:**

Wykład: Powtórzenie podstaw języka C ++, m.in. klas, inicjalizacja obiektów, konstruktory i destruktory, jawne wywołanie, dostęp do treści, przestrzenie nazw, rzutowanie, metody i operatory. Wprowadzenie do nowych elementów C ++, referencji prawostronnych r-value, konstruktorów przenoszenia, uogólnionych wyrażeń stałych, list inicjalizujących, wnioskowania o typach, pętli for bazujących na zapisie z zakresem, a także innych nowych funkcjonalności C++. Biblioteka OpenCV. Kontenery danych, funkcje używane do widzenia komputerowego. Pokazanie i omówienie przykładowych programów.
Zajęcia komputerowe: Analiza danych, głównie obrazów, w środowisku MATLAB. Proste projekty oparte na tym środowisku z interfejsem użytkownika. LabView.
Projekt: Zrealizowany indywidualnie przez studenta projekt, który ma na celu stworzenie prostego programu, w którym wykorzystuje się przedstawione na wykładzie idee programistyczne. Tworzone mogą być zarówno proste programy ilustrujące zagadnienia związane z nowymi funkcjami C++, a także te bardziej skomplikowane, w których np. do przetwarzania obrazu można wykorzystać bibliotekę OpenCV.

**Metody oceny:**

Sprawdzian teoretyczny kończący wykład, prace domowe i sprawdzian praktyczny kończący zajęcia komputerowe, poprawnie opracowany i działający program przedstawiony na zakończenie projektu indywidualnego. Ostateczny wynik jest obliczany jako średnia ważona z testu końcowego (30%), zajęć komputerowych (30%) i projektu (40%). Projekt ocenia dokumentację projektową, kod źródłowy programu, a następnie ocenia działanie samego programu, jego niezawodność i poprawność wyników, które program generuje.

**Egzamin:**

nie

**Literatura:**

1. P. Deitel, H. Deitel, C++ How to Program, 9th Edition, Prentice Hall, 2013
2. B. Stroustrup, The C++ Programming Language, 4th Edition, Addison- Wesley, 2013
3. B. Stroustrup, Programming: Principles and Practice Using C++, 2nd Edition, Addison-Wesley Professional, 2014
4. S. Lippman, J. Lajoie, B. Moo, C++ Primer (5th Edition), Addison-Wesley Professional; 5th edition, 2012

**Witryna www przedmiotu:**

brak

**Uwagi:**

## Charakterystyki przedmiotowe

### Profil ogólnoakademicki - wiedza

**Charakterystyka IPSM1\_W01:**

Znajomość zagadnień z obszaru programowania obiektowego C++

Weryfikacja:

Zaliczenie kolokwium w trakcie wykładu

**Powiązane charakterystyki kierunkowe:** K\_W04, K\_W05

**Powiązane charakterystyki obszarowe:** P6U\_W, I.P6S\_WG.o

**Charakterystyka IPSM1\_W02:**

Zna podstawowe techniki projektowania (UML) i dokumentacji (DOXYGEN) programów komputerowych tworzonych w językach wyższego poziomu C++.

Weryfikacja:

Zaliczenie kolokwium w trakcie wykładu

**Powiązane charakterystyki kierunkowe:** K\_W04, K\_W05

**Powiązane charakterystyki obszarowe:** P6U\_W, I.P6S\_WG.o

### Profil ogólnoakademicki - umiejętności

**Charakterystyka IPSM1\_U01:**

Potrafi zaprojektować i zaimplementować algorytmy w języku obiektowym C++

Weryfikacja:

Zaliczenie projektu programistycznego C++

**Powiązane charakterystyki kierunkowe:** K\_U01, K\_U04, K\_U05, K\_U18

**Powiązane charakterystyki obszarowe:** I.P6S\_UW.o, I.P6S\_UK, P6U\_U, I.P6S\_UO, I.P6S\_UU, III.P6S\_UW.o

**Charakterystyka IPSM1\_U02:**

Potrafi opracować projekt (UML) aplikacji i dokumentację (DOXYGEN) kodów źródłowych

Weryfikacja:

Zaliczenie projektu programistycznego C++

**Powiązane charakterystyki kierunkowe:** K\_U02, K\_U03, K\_U04

**Powiązane charakterystyki obszarowe:** P6U\_U, I.P7S\_UW.o, I.P6S\_UK, I.P6S\_UW.o

### Profil ogólnoakademicki - kompetencje społeczne

**Charakterystyka IPSM1\_S01:**

Potrafi zaprojektować, zaimplementować, udokumentować i zaprezentować swój projekt programistyczny

Weryfikacja:

Zaliczenie projektu programistycznego C++

**Powiązane charakterystyki kierunkowe:** K\_K01, K\_K04

**Powiązane charakterystyki obszarowe:** P6U\_K, I.P6S\_KK, I.P6S\_KO, I.P6S\_KR