



## KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Technologie informacyjne

### Przedmiot

Kierunek studiów

Mechatronika

Studia w zakresie (specjalność)

-

Poziom studiów

pierwszego stopnia

Forma studiów

niestacjonarne

Rok/semestr

1/1

Profil studiów

ogólnoakademicki

Język oferowanego przedmiotu

polski

Wymagalność

obligatoryjny

### Liczba godzin

Wykład

12

Laboratoria

12

Inne (np. online)

0

Ćwiczenia

0

Projekty/seminaria

0

### Liczba punktów ECTS

3

### Wykładowcy

Odpowiedzialny za przedmiot/wykładowca:

prof. dr hab.inż. Wojciech Szelaąg,

Odpowiedzialny za przedmiot/wykładowca:

Wydział Automatyk, Robotyki i Elektrotechniki,

ul. Piotrowo 3, 60-965 Poznań,

pokój 611

### Wymagania wstępne

Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę w zakresie informatyki, matematyki, sprzętu komputerowego, obsługi komputera, systemu operacyjnego Windows oraz podstawowego oprogramowania użytkowego. Powinien również posiadać umiejętność pozyskiwania informacji ze wskazanych źródeł oraz mieć gotowość do podjęcia współpracy w ramach zespołu.

### Cel przedmiotu

Przekazanie podstawowej wiedzy z zakresu informatyki, budowy i działania mikrokomputerów, opanowanie umiejętności opracowywania algorytmów i podstaw programowania w języku C++ i języku HTML oraz umiejętności wykorzystywania oprogramowania użytkowego i zasobów Internetu.

### Przedmiotowe efekty uczenia się

Wiedza



Ma uporządkowaną i podbudowaną teoretycznie wiedzę ogólną w zakresie kluczowych zagadnień technologii informatycznych niezbędnych dla inżyniera mechatronika.

#### Umiejętności

Student umie posługiwać się oprogramowaniem użytkowym oraz korzystać z baz danych i zasobów Internetu, formułować proste algorytmy i programy w języku C++, wykorzystywać technologie informacyjne w praktyce inżyniera.

#### Kompetencje społeczne

Student rozumie potrzebę uczenia się przez całe życie; potrafi inspirować i organizować proces uczenia się innych osób; potrafi współdziałać i pracować w grupie, przyjmuje w niej różne role; potrafi określić priorytety służące realizacji określonego zadania.

#### Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Wykład. Ocena wiedzy i umiejętności na pisemnym kolokwium zaliczeniowym o charakterze łączonym testowym i problemowym. Uzyskiwanie punktów dodatkowych za aktywność podczas wykładów, a szczególnie za: przygotowywanie odpowiedzi na pytania i zadania problemowe podawane przez wykładającego, staranność estetyczną zadań opracowywanych w ramach nauki własnej, aktywność na wykładach przy rozwiązywaniu bieżących zadań problemowych.

Laboratorium. Sprawdzian i premiowanie wiedzy niezbędnej do realizacji kolejnych ćwiczeń laboratoryjnych, ocenianie ciągle, na każdych zajęciach aktywności studenta i przyrostu jego wiedzy oraz umiejętności, a także kompetencji społecznych związanych z pracą w zespole, ocena wiedzy i umiejętności związanych z realizacją określonego ćwiczenia laboratoryjnego, ocena sprawozdania z wykonanego ćwiczenia. Punkty dodatkowych za aktywność podczas zajęć.

#### Treści programowe

Historia informatyki, obszary jej zastosowań i badań. Kodowanie informacji, podstawy działania układów cyfrowych, struktura systemu komputerowego. Systemy operacyjne, praca komputerów w sieci. Internet, intranet. Algorytmy i struktury danych. Wybrane algorytmy rozwiązywalnych analitycznych problemów z matematyki, fizyki oraz algorytmy problemu sortowania. Wstęp do języków programowania. Podstawy programowania w języku C++. Wykorzystanie w praktyce inżyniera wybranych pakietów programów użytkowych.

#### Metody dydaktyczne

Zastosowane metody kształcenia: a) wykład z prezentacją multimedialną (w tym: rysunki, zdjęcia, animacje) uzupełniany przykładami podawanymi na tablicy, b) wykład prowadzony w sposób interaktywny z formułowaniem pytań do studentów, c) uwzględnienie aktywności studentów w czasie zajęć przy wystawianiu oceny końcowej, d) inicjowanie dyskusji w trakcie ćwiczeń laboratoryjnych.

#### Literatura



Podstawowa

1. Cormen T., Leiserson C., Rivest R., Wprowadzenie do algorytmów, WNT, Warszawa, 2007.
2. Grębosz J., Symfonia C++ standard: programowanie w języku C++ orientowane obiektowo. T. 1/2, Instytut Fizyki Jądrowej im. H. Niewodniczańskiego, Polska Akademia Nauk, Kraków, 2006.
3. Metzger P., Anatomia PC, Helion, 2007.
4. Matulewski J., Visual Studio 2013, Helion 2013.

Uzupełniająca

1. Wróblewski P., Algorytmy, struktury danych i techniki programowania, Helion 2015.
2. Stasiewicz A., Ćwiczenia C++11 Nowy standard, Helion, 2012.
3. Wojtuszkiewicz K., Urządzenia techniki komputerowej. Cz.1. Jak działa komputer, PWN, 2011.

**Bilans nakładu pracy przeciętnego studenta**

	Godzin	ECTS
Łączny nakład pracy	75	3,0
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	45	2,0
Praca własna studenta (studia literaturowe, przygotowanie do zajęć, przygotowanie do kolokwium) <sup>1</sup>	30	1,0

<sup>1</sup> niepotrzebne skreślić lub dopisać inne czynności