

<b>KARTA OPISU MODUŁU KSZTAŁCENIA</b>		
Nazwa modułu/przedmiotu <b>Informatyka</b>		Kod <b>1010804111010820031</b>
Kierunek studiów <b>Elektronika i Telekomunikacja</b>	Profil kształcenia (ogólnoakademicki, praktyczny) <b>ogólnoakademicki</b>	Rok / Semestr <b>1 / 1</b>
Ścieżka obieralności/specjalność <b>-</b>	Przedmiot oferowany w języku: <b>polski</b>	Kurs (obligatoryjny/obieralny) <b>obligatoryjny</b>
Stopień studiów: <b>I stopień</b>	Forma studiów (stacjonarna/niestacjonarna) <b>niestacjonarna</b>	
Godziny Wykłady: <b>25</b> Ćwiczenia: <b>-</b> Laboratoria: <b>20</b> Projekty/seminaria: <b>-</b>		Liczba punktów <b>6</b>
Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) <b>kierunkowy</b>		(ogólnouczelniany, z innego kierunku) <b>ogólnouczelniany</b>
Obszar(y) kształcenia i dziedzina(y) nauki i sztuki <b>nauki techniczne</b> <b>nauki techniczne</b>		Podział ECTS (liczba i %) <b>6 100%</b> <b>6 100%</b>
<b>Odpowiedzialny za przedmiot / wykładowca:</b> dr hab. inż. Mariusz Głabowski, prof. nadzw. email: mariusz.glabowski@put.poznan.pl tel. +48 61 665 3904 Wydział Elektroniki i Telekomunikacji ul. Piotrowo 3A 60-965 Poznań		
<b>Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych:</b>		
1	<b>Wiedza:</b>	Podstawowe wiadomości z logiki matematycznej. K1_W01
2	<b>Umiejętności:</b>	Potrąfi pozyskiwać informacje z literatury oraz innych źródeł w języku polskim lub angielskim; potrafi integrować uzyskane informacje, dokonywać ich interpretacji i wyciągać wnioski. K1_U01
3	<b>Kompetencje społeczne</b>	Zna ograniczenia własnej wiedzy i umiejętności, rozumie konieczność dalszego dokształcania się. K1_K01
<b>Cel przedmiotu:</b> Zapoznanie z podstawami algorytmów i struktur danych, złożonością obliczeniową oraz zasadami tworzenia programów komputerowych. Zapoznanie z językiem obiektowym C++.		
<b>Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia</b>		
<b>Wiedza:</b> 1. Zna zasady konstrukcji programów komputerowych, posiada wiedzę z zakresu informatyki i zna składnię języków oprogramowania C, C++, - [K1_W09] 2. Ma podstawową wiedzę o algorytmach (sortowania, przeszukiwania zbiorów danych, metod zachłanych) i struktur danych (tablice, drzewa binarne, grafy i metody ich przeszukiwania) - [[K1_W09]]		
<b>Umiejętności:</b> 1. Potrafi programowo zrealizować podstawowe algorytmy obliczeniowe za pomocą popularnych języków programowania - [K1_U13] 2. Potrafi się posługiwać językami programowania wysokiego poziomu C, C++. - [K1_U13] 3. Potrafi pisać i uruchamiać programy pozwalające rozwiązywać wybrane problemy techniczne związane z telekomunikacją - [K1_U13]		
<b>Kompetencje społeczne:</b> 1. Zna ograniczenia własnej wiedzy i umiejętności, rozumie konieczność dalszego dokształcania się - [K1_K01] 2. Posiada świadomość konieczności profesjonalnego podejścia do rozwiązywanych problemów technicznych i podejmowania odpowiedzialności za proponowane przez siebie rozwiązania techniczne - [K1_K02]		
<b>Sposoby sprawdzenia efektów kształcenia</b>		

<p>Wykład                  - pisemny egzamin</p> <p>Laboratoria:                  - sprawdzian przyrostu wiedzy niezbędnej do realizacji postawionych problemów w danym obszarze tematyki przedmiotu;                  - ocenianie ciągle, na każdych zajęciach - premiowanie przyrostu umiejętności posługiwania się poznanymi zasadami i metodami.</p>		
<b>Treści programowe</b>		
<ul style="list-style-type: none"> <li>- architektura komputerów</li> <li>- złożoność obliczeniowa</li> <li>- algorytmy rekurencyjne i zachłanne</li> <li>- struktura programu w języku C</li> <li>- podstawowe typy danych</li> <li>- operatory i wyrażenia</li> <li>- instrukcje sterujące</li> <li>- tablice</li> <li>- funkcje</li> <li>- przekazywanie argumentów</li> <li>- wzorce funkcji</li> <li>- algorytmy sortowania i przeszukiwania</li> <li>- wskaźniki i dynamiczne przydzielanie pamięci</li> <li>- struktura programu w C++</li> <li>- klasy i obiekty klas</li> <li>- dziedziczenie</li> <li>- polimorfizm</li> <li>- wzorce klas</li> <li>- programowanie zorientowane obiektowo</li> </ul>		
<p><b>Literatura podstawowa:</b></p> <ol style="list-style-type: none"> <li>1. D.E. Knuth, The art of computer programming, Addison-Wesley Publishing Company, Reading, MA, 1968, 1973.</li> <li>2. N. Wirth, Algorytmy + struktury danych = programy, WNT, Warszawa, 1980</li> <li>3. T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, Wprowadzenie do algorytmów, WNT, Warszawa, 2004.</li> <li>4. J. Grębosz, Symfonia C++, Oficyna Kallimach, Kraków 2008.</li> </ol>		
<p><b>Literatura uzupełniająca:</b></p>		
<b>Bilans nakładu pracy przeciętnego studenta</b>		
<b>Czynność</b>		<b>Czas (godz.)</b>
1. Wykłady		25
2. Laboratoria		20
3. Indywidualne pisanie programów		40
4. Przygotowanie do laboratorów		40
5. Przygotowanie do egzaminu		30
6. Egzamin		3
7. Konsultacje		2
<b>Obciążenie pracą studenta</b>		
<b>forma aktywności</b>	<b>godzin</b>	<b>ECTS</b>
Łączny nakład pracy	160	6
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	50	2
Zajęcia o charakterze praktycznym	100	4