

**SYLABUS**DOTYCZY CYKLU KSZTAŁCENIA 2021-2024  
(skrajne daty)

Rok akademicki 2021/2022

**1. PODSTAWOWE INFORMACJE O PRZEDMIOCIE**

Nazwa przedmiotu	<b>Podstawy programowania</b>
Kod przedmiotu*	
Nazwa jednostki prowadzącej kierunek	Kolegium Nauk Przyrodniczych
Nazwa jednostki realizującej przedmiot	Kolegium Nauk Przyrodniczych Instytut Informatyki
Kierunek studiów	Matematyka
Poziom studiów	studia I stopnia
Profil	ogólnoakademicki
Forma studiów	stacjonarne
Rok i semestr studiów	rok I, semestr 2
Rodzaj przedmiotu	kierunkowy
Język wykładowy	język polski
Koordinator	dr Paweł Drygaś
Imię i nazwisko osoby prowadzącej / osób prowadzących	dr Paweł Drygaś

\* - zgodnie z ustaleniami w Jednostce

**1.1. Formy zajęć dydaktycznych, wymiar godzin i punktów ECTS**

Semestr (nr)	Wykł.	Ćw.	Konw.	Lab.	Sem.	ZP	Prakt.	Inne (jakie?)	Liczba pkt ECTS
2	30			30					6

**1.2. Sposób realizacji zajęć**

- zajęcia w formie tradycyjnej
- zajęcia realizowane z wykorzystaniem metod i technik kształcenia na odległość

**1.3 Forma zaliczenia przedmiotu /modułu (z toku) (egzamin, zaliczenie z oceną, zaliczenie bez oceny)****Wykład-Egzamin, Ćwiczenia - zaliczenie na ocenę****2. WYMAGANIA WSTĘPNE**

Podstawowe wiadomości z zakresu budowy i obsługi komputera.
---

### 3. CELE, EFEKTY UCZENIA SIĘ, TREŚCI PROGRAMOWE I STOSOWANE METODY DYDAKTYCZNE

#### 3.1 Cele przedmiotu

C <sub>1</sub>	Przygotowanie do rozwiązywania problemów (również matematycznych) przy użyciu środków informatyki oraz korzystania z komputerów na zajęciach matematycznych.
C <sub>2</sub>	Wprowadzenie w zagadnienia algorytmicznego opisu rozwiązywania problemów – poszukiwanie, dostrzeganie i konstruowanie algorytmów.
C <sub>3</sub>	Zapoznanie studentów z podstawowymi metodami projektowania algorytmów, pisania, uruchamiania i testowania programów. Analiza wyników obliczeń

#### 3.2 Efekty uczenia się dla przedmiotu

EK (efekt uczenia się)	Treść efektu uczenia się zdefiniowanego dla przedmiotu	Odniesienie do efektów kierunkowych
EK_01	student zna podstawy technik obliczeniowych i programowania które mogą być wykorzystane we wspomaganii pracy matematyka oraz rozumie ograniczenia pojawiające się przy ich wykorzystaniu	K_W05
EK_02	student potrafi rozpoznać i dokonać specyfikacji problemu, który można rozwiązać algorytmicznie, wyszukuje w/w problemy w oparciu o zagadnienia praktyczne	K_U14
EK_03	student dla danego problemu potrafi ułożyć i przeanalizować algorytm zgodny ze specyfikacją i zapisać go w odpowiednim języku programowania, a następnie testować napisany samodzielnie program komputerowy i dokonywać niezbędnych korekt	K_U14
EK_04	student wyraża własne opinie na temat teoretycznych i praktycznych zagadnień z algorytmiki dotyczących danego problemu oraz formułuje pytania służące zrozumieniu badanego problemu, a w razie potrzeby uzupełnia swoje kompetencje.	K_K01
EK_05	student jest gotów do prezentowania krytycznej postawy wobec odbieranych treści, ma świadomość błędów, które mogą się pojawić przy układaniu algorytmów oraz zapisach w używanych językach programowania,	K_K02
EK_06	student jest gotów do uznawania znaczenia wiedzy dotyczącej układania algorytmów w rozwiązywaniu problemów praktycznych.	K_K03

### 3.3 Treści programowe

#### A. Problematyka wykładu

Treści merytoryczne
<b>Wiadomości wstępne.</b> Pojęcie informatyki, informacji i jej przetwarzania. Dziedziny informatyki
<b>Algorytmy.</b> Pojęcie algorytmu. Interpretacja zmiennych w algorytmie, zapis algorytmów. Schematy blokowe (sieci działań). Testowanie algorytmu. Klasyfikacja prostych algorytmów
<b>Komputery.</b> Schemat funkcjonalny komputera, organizacja pamięci, arytmometr, układ sterowania, język wewnętrzny. Arytmetyka komputerów
<b>Języki algorytmiczne.</b> Pojęcie języka. Gramatyki kombinatoryczne. Języki programowania
<b>Budowa programu na przykładzie języka C++.</b> Związek programu z algorytmem. Stałe i zmienne. Podstawowe instrukcje.
<b>Funkcje standardowe i programowanie wyrażeń.</b> Dziedzina i zbiór wartości funkcji standardowych. Operacje arytmetyczne i logiczne. Gramatyka i semantyka wyrażeń
<b>Instrukcje złożone.</b> Instrukcje warunkowe. Różne rodzaje pętli
<b>Zmienne złożone.</b> Typ tablicowy, Typ plikowy. Operacje na zmiennych w/w typów
<b>Podprogramy</b> Idea podprogramu, funkcje. Wymiana danych pomiędzy programem a podprogramem

#### B. Problematyka ćwiczeń audytoryjnych, konwersatoryjnych, laboratoryjnych, zajęć praktycznych

Treści merytoryczne
<b>Algorytmy.</b> Zapis algorytmiczny rozwiązania prostych problemów. Różne sposoby zapisu algorytmów - schematy blokowe. Testowanie algorytmu.
<b>Budowa programu na przykładzie języka C++.</b> Związek programu z algorytmem. Stałe i zmienne. Podstawowe instrukcje. Tworzenie prostych programów z użyciem operacji arytmetycznych, logicznych i standardowych funkcji oraz ich testowanie i wyciąganie wniosków.
<b>Instrukcje złożone.</b> Instrukcje warunkow IF. Użycie pętli Do, While i For oraz pokazanie na przykładach istotnych różnic.
<b>Zmienne złożone.</b> Użycie zmiennych typu tablicowego i plikowego. Operacje na w/w zmiennych
<b>Podprogramy.</b> Tworzenie prostych funkcji i ich użycie w programie. Używanie zmiennych globalnych i lokalnych. Iteracja.

### 3.4 Metody dydaktyczne

Wykład z prezentacją multimedialną : wykład problemowy

Laboratorium: rozwiązywanie zadań, dyskusja, projektowanie algorytmów, tworzenie programów.

## 4. METODY I KRYTERIA OCENY

### 4.1 Sposoby weryfikacji efektów uczenia się

Symbol efektu	Metody oceny efektów uczenia się (np.: kolokwium, egzamin ustny, egzamin pisemny, projekt, sprawozdanie, obserwacja w trakcie zajęć)	Forma zajęć dydaktycznych (w, ćw, ...)
EK_01	egzamin, kolokwium	wykład, laboratorium
EK_02	egzamin, kolokwium	wykład, laboratorium
EK_03	egzamin, kolokwium	wykład, laboratorium
EK_04	obserwacja w trakcie zajęć	wykład, laboratorium
EK_05	obserwacja w trakcie zajęć	wykład, laboratorium
EK_06	obserwacja w trakcie zajęć	wykład, laboratorium

### 4.2 Warunki zaliczenia przedmiotu (kryteria oceniania)

Wykład – egzamin

Egzamin pisemny teoretyczno-zadaniowy

Oceny wystawiane są według następującej skali

0 – 50	niedostateczny
50 – 60	dostateczny
60 – 70	plus dostateczny
70 – 80	dobry
80 – 90	plus dobry
90 – 100	bardzo dobry

W przypadku niespełnienia powyższych warunków student może przystąpić do egzaminu poprawkowego ocenianego zgodnie z powyższymi zasadami

Warunkiem koniecznym uzyskania zaliczenia z laboratorium jest uzyskanie minimum 30 % punktów z każdego z kolokwiów. W przypadku niespełnienia powyższych warunków student może dwukrotnie przystąpić do kolokwium poprawkowego

Aktywność podczas zajęć premiowana jest dodatkowymi punktami (maksymalnie 5 punktów).

Oceny wystawiane są według następującej skali

0 – 50	niedostateczny
50 – 60	dostateczny
60 – 70	plus dostateczny
70 – 80	dobry

80 – 90	plus dobry
90 – 100	bardzo dobry

## 5. CAŁKOWITY NAKŁAD PRACY STUDENTA POTRZEBNY DO OSIĄGNIĘCIA ZAŁOŻONYCH EFEKTÓW W GODZINACH ORAZ PUNKTACH ECTS

Forma aktywności	Średnia liczba godzin na zrealizowanie aktywności
Godziny kontaktowe wynikające z harmonogramu studiów	60
Inne z udziałem nauczyciela (udział w konsultacjach, egzaminie)	5
Godziny niekontaktowe – praca własna studenta (przygotowanie do zajęć, egzaminu, napisanie referatu itp.)	85
SUMA GODZIN	150
SUMARYCZNA LICZBA PUNKTÓW ECTS	6

\* Należy uwzględnić, że 1 pkt ECTS odpowiada 25-30 godzin całkowitego nakładu pracy studenta.

## 6. PRAKTYKI ZAWODOWE W RAMACH PRZEDMIOTU

wymiar godzinowy	nie dotyczy
zasady i formy odbywania praktyk	nie dotyczy

## 7. LITERATURA

<p>Literatura podstawowa:</p> <ol style="list-style-type: none"> <li>1. D. Harel, Rzecz o istocie informatyki: algorytmika, WNT, Warszawa 2008.</li> <li>2. N. Wirth, Wstęp do programowania systematycznego, WNT, Warszawa 1987.</li> <li>3. N. Wirth, Algorytmy+struktury danych=programy, WNT, Warszawa 2004.</li> <li>4. J. Liberty, Poznaj C++ w 24 godziny, Intersoftland, Warszawa 1999.</li> </ol>
<p>Literatura uzupełniająca:</p> <ol style="list-style-type: none"> <li>1. A.V. Aho, J.E. Hopcroft i J.D. Ullman, Algorytmy i struktury danych, Helion, Warszawa 2003</li> <li>2. L. Banachowski, K. Diks i W. Rytter, Algorytmy i struktury danych, WNT, Warszawa 2006.</li> <li>3. T.H. Cormen, Ch.E. Leiserson, R.L. Rivest i C. Stein, Wprowadzenie do algorytmów, WNT, Warszawa 2007.</li> </ol>

4. M. Sysło, Algorytmy, WSiP, Warszawa 2008.

5. <http://wazniak.mimuw.edu.pl>

Akceptacja Kierownika Jednostki lub osoby upoważnionej