

Wstęp do Informatyki 1.1

wykład 30 godz. ćwiczenia 15 godz. laboratorium 15 godz.

1. Wprowadzenie (1 wykład)
 - (a) Co to jest informatyka – krótka historia.
 - (b) Pojęcie zadania, danych, wyniku i algorytmu.
 - (c) Budowa komputera. System operacyjny i jego zadania (w lab. zapoznanie się z Linuxem i programami powłoki jądra Unix). Zasoby komputera. Graficzne środowiska pracy.
2. Od problemu do programu (2 wykłady)
 - (a) Zadania i algorytmy. Opis słowny algorytmu. Przykłady zadań i algorytmów (np. proste sortowania).
 - (b) Kod źródłowy, kompilacja, program wykonywalny.
3. Arytmetyka zmiennopozycyjna (2 wykłady)
 - (a) Reprezentacja liczb w komputerze. Standard IEEE-754.
 - (b) Proste algorytmy numeryczne (np. sumowanie, schemat Hornera, mnożenie macierzy). Błędy zaokrągleń.
4. Wprowadzenie do programowania (5 wykładów)
 - (a) Podstawowe konstrukcje języka programowania imperatywnego (zalecany jest język C).
 - (b) Proste typy danych, deklaracje zmiennych.
 - (c) Instrukcja przypisania, instrukcje warunkowe, instrukcja złożona.
 - (d) Iteracja - instrukcje for, while.
 - (e) Złożone typy danych: tablice, struktury.
 - (f) Procedury i funkcje, przekazywanie parametrów, deklaracje lokalne i globalne, zasięg widoczności.
 - (g) Pliki. Procedury wejścia/wyjścia.
 - (h) Poprawność programu (częściowa, całkowita, metoda niezmienników).
5. Rekurencja (2 wykłady)
 - (a) Proste algorytmy rekurencyjne.
 - (b) Zasada „dziel i rządź”. Algorytmy MergeSort/QuickSort.

Wstęp do Informatyki 1.2

wykład 30 godz. ćwiczenia 15 godz. laboratorium 15 godz.

1. Elementy analizy złożoności obliczeniowej (3 wykłady)
 - (a) Rozmiar zadania. Złożoność czasowa i pamięciowa.
 - (b) Rząd złożoności. Wpływ rzędu złożoności na praktyczną przydatność algorytmu.
 - (c) Obliczanie złożoności prostych algorytmów (sortowanie i wyszukiwanie binarne).
 - (d) Obliczanie złożoności algorytmów rekurencyjnych (pamięciowej i czasowej).
2. Abstrakcyjne struktury danych i metody ich implementacji (4 wykłady)
 - (a) Stosy, kolejki, kolejki priorytetowe. Przykłady użycia (algorytm Heap-Sort). Implementacja tablicowa.
 - (b) Wskaźniki. Dynamiczna alokacja pamięci.
 - (c) Listy. Listowa implementacja stosu, kolejki i kolejki priorytetowej.
 - (d) Drzewa binarnych wyszukiwań.
3. Grafy i algorytmy grafowe (4 wykłady)
 - (a) Grafy i ich reprezentacje.
 - (b) Podstawowe algorytmy grafowe. Przeszukiwanie włąb i wszere.
4. Informacja o problemach NP-zupełnych i nierozstrzygalnych. (1 wykład)