

Zajęcia 13 (2 godziny)

Temat: Algorytmy sortowania, funkcja sort z biblioteki <algorithm>, własna funkcja porównująca sortowane elementy.

Treści z sylabusu:

Matematyka:	Programowanie:	Algorytmika
	Funkcje rekurencyjne, zasięg zmiennych, sortowanie w bibliotece algorithm	Sortowanie szybkie, sortowanie przez scalanie, sortowanie stabilne, inwersje w tablicy

Zadanie sprawdzające opanowanie poprzedniej lekcji

- Zmodyfikuj program zawierający funkcję z sortowaniem szybkim tak, by na ekranie wyświetlane były kolejno parametry z wywołań rekurencyjnych funkcji. Narysuj w zeszycie drzewo wywołań rekurencyjnych funkcji.

Czynności nauczyciela:

- Przypomina działanie poznanych algorytmów sortowania
- omawia wybrane funkcje z biblioteki <algorithm>: sort, stable_sort, reverse, rotate, random_shuffle, merge, itp
- opisuje sposób modyfikowania funkcji sort, zmieniający sposób sortowania danych
- wyjaśnia sposób zapisu i zastosowania własnych komparatorów i funkcji porównujących
- pokazuje sposób działania ww. funkcji i algorytmów

Przykładowe zadania do rozwiązania na lekcji:

- Latarnie – podane są w przypadkowej kolejności położenia latarni wzdłuż drogi. Znając długość drogi i położenia latarni należy znaleźć minimalny zasięg latarni, by cała droga była oświetlona.

Przykładowe zadania domowe:

- Napisz program, który posortuje kątowo n punktów
- Posortuj leksykograficznie listy uczniów – należy wziąć pod uwagę klasę, nazwisko i imiona (max 2) ucznia

Zadania na sprawdzarce:

- Bierki – ze zbioru bierek o podanej długości należy zostawić maksymalnie wiele takich, że z każdej trójki da się zbudować trójkąt.
- Iloczyn – należy obliczyć maksymalny iloczyn, jaki można uzyskać mnożąc przez siebie dowolne trzy (możliwe że takie same) liczby ze zbioru zawierającego liczby dodatnie i ujemne.
- Gwoździe – mając informacje o wysokości gwoździ i liczbie uderzeń młotkiem należy znaleźć największą możliwą liczbę gwoździ, które będą wystawały ponad powierzchnie na taką samą wysokość

Uwagi do realizacji:

Przy okazji omawiania funkcji z biblioteki <algorithm> można pokazać ciekawe (samodzielne) implementacje wybranych funkcji w C++