

## Projekt – sortowanie

Należy zaimplementować, przetestować i porównać metody sortowania tablic.

I grupa metod:

- Przez wstawianie
- Przez selekcję
- Sortowanie bąbelkowe

II grupa metod:

- Quicksort
- Sortowanie Shella
- Sortowanie przez kopcowanie

Sortowaniu podlegają tablice o wartościach typu int z zakresu  $<-100; 100>$ . Należy przeprowadzić trzy rodzaje testów sortowania w kolejności rosnącej dla każdej z metod, w zależności od przygotowania danych wejściowych.

- Test1 - Dla danych wygenerowanych losowo
- Test2 - Dla danych posortowanych w kolejności odwrotnej (malejąco)
- Test3 - Dla danych posortowanych właściwie (rosnąco)

Dane posortowane wczytujemy z pliku o nazwie **dane** (uzyskujemy go własnym programem sortującym, który zapisuje do niego wyniki).

Analiza wyników dla każdego testu (Test1 – Test3) powinna zawierać wykresy zależności czasu sortowania od rozmiaru tablicy

- jeden wykres zbiorczy dla 3 metod podstawowych,
  - jeden wykres zbiorczy dla II grupy metod,
  - jeden wykres porównujący wszystkie metody I i II grupy
- ✓ w sumie 9 wykresów – po 3 dla każdego zestawu danych.

Do badania czasów można wykorzystać funkcję systemową *time* lub *gettimeofday*. Wywołujemy ją w programie tuż przed i tuż po sortowaniu i bierzemy pod uwagę różnicę uzyskanych czasów.

Maksymalny rozmiar tablicy oraz długość listy dobieramy tak, by czas sortowania nie przekraczał 15 minut. **Program pyta użytkownika o rozmiar tablicy do sortowania.**

Należy przedstawić algorytmy (pseudokody lub schematy blokowe metod) oraz ich złożoności obliczeniowe porównane z uzyskanymi wynikami (próba aproksymacji otrzymanych wykresów odpowiednimi funkcjami złożonościowymi).

Sprawozdanie w wersji elektronicznej (plik w formacie pdf) – proszę oddać poprzez platformę Delta, a programy w wersji elektronicznej proszę zapisać na serwerze torus we wskazanym katalogu. Ma to być jeden plik – archiwum tar o nazwie: **nazwiskoimie.tar (np.kowalskijan.tar)**. W pliku – archiwum powinny się znaleźć pliki źródłowe programu oraz plik **Makefile**, który wskaże programowi make

sposób kompilacji programu. Plik z danymi nazywa się dane i znajduje się w tym katalogu, co program.

Archiwum można utworzyć poleceniem:

```
tar cvf nowakjan.tar algorytmy.h main.c algorytmy.c Makefile
```

Przykładowy plik Makefile oraz zarys programu jest na platformie Delta.

**Proszę przetestować wysyłane programy na serwerze torus (razem z plikiem Makefile).**

W programie wysłanym na torus mają się znaleźć funkcje sortujące oraz minimalna funkcja main, w której użytkownik wprowadza rozmiar tablicy, a program wczytuje dane z pliku dane i wywołuje wszystkie funkcje sortujące. Sposób przeprowadzenia wszystkich testów proszę sobie zorganizować w sposób dowolny (przygotowanie pliku z danymi, zbieranie czasów, przetworzenie wyników).