

## Języki Programowania, 8.10.2011

W języku C napisać program sortujący, używający sortowania bąbelkowego. Należy posortować tablicę o MAX elementach wypełnioną liczbami losowymi.

Program musi zawierać funkcje:

wypełnienie tablicy statycznej 7 elementowej liczbami losowymi **1 p**

`void swap(int *a, int *b)` - funkcja zamieniająca dwa elementy: *a* i *b* **1 p**

`void sort(int *tab)` - funkcja sortująca, sortuje tablicę *tab* używając sortowania bąbelkowego **1 p**

`void print(int *tab)` - funkcja wypisująca na ekran tablicę *tab* **1 p**

działający program sortujący tablicę **1 p**

**dodatkowo:** zamienić tablicę statyczną na tablicę z dynamiczną alokacją pamięci

### Sortowanie bąbelkowe:

W trakcie sortowania bąbelkowego następują przejścia po wszystkich elementach tablicy, w których porównywane są kolejne elementy (dwa sąsiadujące). Jeśli elementy są w złej kolejności to są one zamieniane miejscami. Przy każdej iteracji (przejściu) poprawnie zostaje ustawiony co najmniej jeden, ostatni (najwyższy) element tablicy.

Wersja bez optymalizacji, przykład działania:

Przejście	Tablica	Opis operacji
1	4 2 1 8 9	4 > 2, konieczne przestawienie elementów (swap)
	2 4 1 8 9	4 > 1, konieczne przestawienie elementów (swap)
	2 1 4 8 9	4 < 8, elementy w dobrej kolejności
	2 1 4 8 9	8 < 9, elementy w dobrej kolejności
	2 1 4 8 9	Koniec pierwszego obiegu, ostatni element odpowiednio ustawiony
2	2 1 4 8 9	2 > 1, konieczne przestawienie elementów (swap)
	1 2 4 8 9	2 < 4, elementy w dobrej kolejności
	1 2 4 8 9	4 < 8, elementy w dobrej kolejności
	1 2 4 8 9	Koniec drugiego obiegu, dwa ostatnie elementy odpowiednio ustawione
3	1 2 4 8 9	1 < 2, elementy w dobrej kolejności
	1 2 4 8 9	2 < 4, elementy w dobrej kolejności
	1 2 4 8 9	Koniec trzeciego obiegu, trzy ostatnie elementy odpowiednio ustawione
4	1 2 4 8 9	1 < 2, elementy w dobrej kolejności
	1 2 4 8 9	Koniec czwartego obiegu, tablica posortowana

Przykład outputu:

Tablica przed sortowaniem:

1 4 8 12 2 3 5

Tablica po sortowaniu:

1 2 3 4 5 8 12

Kompilacja programu:

**g++ -Wall <nazwa-pliku-c> -o <nazwa-skompilowanego-programu>**

np. `g++ -Wall program.c -o program`

-Wall – flaga włączająca ostrzeżenia kompilatora. Program powinien kompilować się bez żadnych ostrzeżeń.