

Politechnika Śląska w Gliwicach
Wydział Informatyki, Elektroniki i Informatyki

Podstawy Programowania Komputerów

Sortowanie

Autor:	Tomasz Korniluk
Prowadzący:	dr inż. Jakub Nalepa
rok akademicki:	2017/2018
kierunek:	informatyka
rodzaj studiów:	SSI
semestr:	1
termin laboratorium / ćwiczeń:	czwartek, 13:00 – 14:30
grupa:	5
sekcja:	14
termin oddania sprawozdania:	2017-12-01
data oddania sprawozdania:	2017-11-30

1. Treść zadania:

Powinny zostać zaimplementowane co najmniej trzy algorytmy sortowania (wykluczamy sortowanie bąbelkowe), np. sortowanie przez wstawianie, sortowanie szybkie, sortowanie przez scalanie, sortowanie przez wybieranie, sortowanie przez kopcowanie itd. Dla szeregów liczbowych zapisanych w pliku tekstowym powinna istnieć możliwość wyboru algorytmu sortowania. Powinna istnieć możliwość zapisania wyników do pliku tekstowego.

2. Analiza zadania:

Zagadnienie przedstawia problem sortowania liczb zapisanych w pliku.

2.1 Algorytmy:

Użytkownik wybiera jeden z 3 zaimplementowanych w programie metod sortowania, plik w którym występują liczby, oraz plik do którego wynik ma być zapisany. Następnie program wczytuje liczby z wiersza do tablicy, aż do napotkania znaku końca „/” i sortuje. Liczby po posortowaniu są wypisywane od razu, następnie program przechodzi dalej (wiersz niżej). Jeśli w wierszu pliku wystąpił niedozwolony znak, lub ponad 2 symbole rozdzielające liczby (spacje) program wypisze błąd. W programie użyto sortowania przez wybór, sortowania przez wstawianie, oraz sortowania zwariowanego (możliwie najgorszy ze sposobów sortowania, im więcej liczb do posortowania tym gorsza praca, ten typ sortowania umieszczono w programie, aby użytkownik mógł zobaczyć, dlaczego ma tak złą sławę).

3 Specyfikacja zewnętrzna:

Program jest uruchamiany z linii poleceń. Należy przekazać do linii poleceń kolejno: nazwę programu, plik wejściowy i plik wyjściowy umieszczonych w tym samym folderze. Np.

Program.exe wczytaj.txt wypisz.txt

Następnie program prosi o podanie typu sortowania. Użytkownik musi podać a, b lub c.

4 Specyfikacja wewnętrzna:

W programie rozdzielono interfejs (komunikacje z użytkownikiem) od logiki aplikacji (sortowania liczb).

4.1 Szczegółowy opis implementacji funkcji:

```
int poprawnosc(std::string c)
```

Funkcja sprawdza poprawność wczytanego wiersza pliku tekstowego. Jeśli napotka błąd daje informację o błędzie, która wykorzystywana jest w funkcji `void glowna(char sorttype, std::string wczytaj, std::string lel)`

```
void glowna(char sorttype, std::string wczytaj, std::string lel)
```

Jest to funkcja główna programu, jako jedyna wywoływana w `int main()` w której deklarowana jest reszta funkcji. Funkcja ta otwiera plik i wczytuje kolejno wiersze sortując je. W przypadku błędu zapamiętuje błąd i przechodzi dalej. Pod koniec wypisuje ilość wierszy i ilość błędnych wierszy.

```
void wybor(int size, double t[])
```

Funkcja sortowania przez wybór wywoływana w funkcji „glowna”. Jeśli wcześniej użytkownik wybrał opcję sortowania przez wybór.

```
void wstawianie(int size, double t[])
```

Funkcja sortowania przez wstawianie wywoływana w funkcji „glowna”. Jeśli wcześniej użytkownik wybrał opcję sortowania przez wstawianie.

```
bool Posortowane(int rozmiar, double t[])
```

Funkcja używana tylko przy wcześniejszym wyborze sortowania zwariowanego. Zwraca prawdę, jeżeli tablica liczb jest posortowana, co później w funkcji głównej informuje program o możliwości wypisania go.

```
void Tasuj(int rozmiar, double t[])
```

Funkcja używana tylko przy wcześniejszym wyborze sortowania zwariowanego. Losowo zmienia kolejność liczb w tablicy.

5 Testowanie:

Program został przetestowany na plikach poprawnych jak i niepoprawnych (zawierający liczby w niepoprawnym formacie, niezgodne ze specyfikacją). Pliki niepoprawne powodują zliczenie błędnych wierszy i poinformowaniu o tym użytkownika, ale program dalej wykonuje sortowanie na poprawnych wierszach. Plik pusty nie powoduje zgłoszenia błędu, ale tworzy pusty plik wyjściowy, z informacją o braku błędów i braku wierszy. Jeśli w pliku cyfry są rozdzielone ponad 2 spacjami wypisuje błąd. Maksymalna liczba akceptowana w pliku zależy od kompilatora (do liczb użyta została zmienna przecinkowa typu `double`). W przypadku gdy plik zwraca 2 (`return 2;`) wypisuje błąd. Błąd ten świadczy o niemożności załokowania pamięci dla tablic dynamicznych użytych w programie.

6. Wnioski:

Program do sortowania nie należy do skomplikowanych programów, choć wymaga myślenia. Głównym problemem zadania było odpowiednie wczytanie plików. Jeśli wczytywano wiersz po wierszu w stringu trzeba było przekonwertować na tablicę znaków char i stworzyć algorytm, który w zależności od napotkanego znaku będzie tworzył liczbę i zapisywał ją do tablicy liczbowej zmiennoprzecinkowej. Po powstaniu tablicy trzeba było ją posortować. Wymagało to wcześniejszego utworzenia funkcji sortujących, wywoływanych w zależności od wyboru użytkownika. Najmniejszym problemem było wyszukiwanie błędów w pliku i zwracania informacji o nim.

7. Literatura:

Zapoznanie z algorytmami sortującymi:

http://eduinf.waw.pl/inf/alg/003_sort/index.php