**SPRAWOZDANIE**

|  |  |  |  |
| --- | --- | --- | --- |
| **Przedmiot** | Wprowadzenie do Informatyki | **Zadanie** | 8.3 |
| **Autor** | Mateusz Jasiński | **Grupa** | WCY20IY2S1 |
| **Temat** | Pliki i strumienie – pliki tekstowe. | | |

1. Treść

Program wczytuje z klawiatury trzy nazwy plików. Dwa pierwsze pliki zawierają posortowane liczby. Program łączy te pliki tak, aby zapisując je do trzeciego pliku w pliku liczby były również posortowane oraz wyświetla na ekranie zawartość trzeciego pliku w wierszach po 10 liczb.

* 1. Metoda realizacji

Odczytujemy z dwóch plików elementy do dwóch tablic, następnie tablice scalamy do jednej tablicy i wypisujemy jej elementy na ekran oraz zapisujemy je do trzeciego pliku.

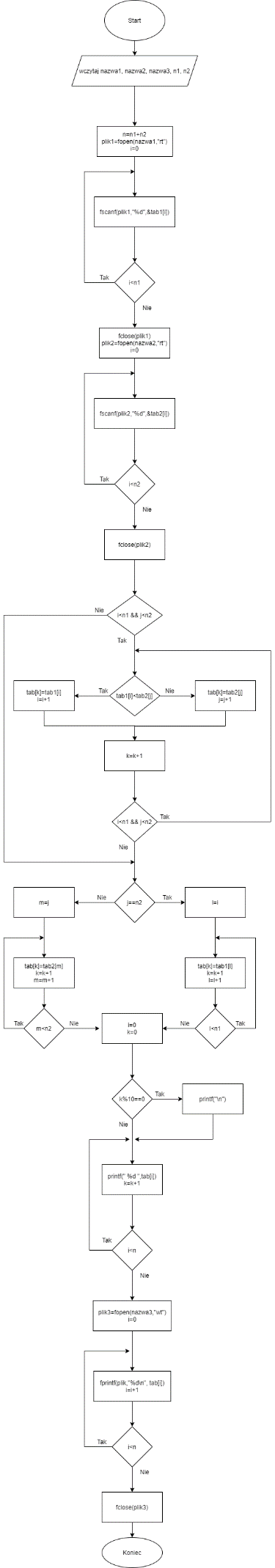
* 1. Założenia / ograniczenia dotyczące danych:
     1. Dane wejściowe

Nazwy trzech plików – wprowadzane z klawiatury.

* + 1. Dane wyjściowe

n posortowanych elementów – wyprowadzane na ekran i zapisane do trzeciego pliku.

1. Realizacja
   1. Algorytm



* 1. Kod źródłowy

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int \*wczytaj(char \*nazwa\_pliku1, char \*nazwa\_pliku2, int \*tab, int \*n) {

FILE\* plik1=fopen(nazwa\_pliku1,"r");

FILE\* plik2=fopen(nazwa\_pliku2,"r");

if(!plik1 || !plik2) {

printf("Nie znaleziono pliku do wczytania!");

exit(0);

}

int i, n1=0, n2=0, \*tab1, \*tab2, x1, x2;

while(fscanf(plik1,"%\*d")!=EOF)

n1++;

while(fscanf(plik2,"%\*d")!=EOF)

n2++;

rewind(plik1);

rewind(plik2);

\*n=n1+n2;

tab=(int\*)malloc(\*n\*sizeof(int));

tab1=(int\*)malloc(n1\*sizeof(int));

tab2=(int\*)malloc(n2\*sizeof(int));

for(i=0;i<n1;i++)

fscanf(plik1,"%d",&tab1[i]);

for(i=0;i<n2;i++)

fscanf(plik2,"%d",&tab2[i]);

for(i=0,x1=0,x2=0;x1<n1||x2<n2;i++) {

if(x1<n1 && x2<n2) {

if(tab1[x1]<tab2[x2]) {

tab[i]=tab1[x1];

x1++;

}

else {

tab[i]=tab2[x2];

x2++;

}

}

else if(x1<n1) {

tab[i]=tab1[x1];

x1++;

}

else if(x2<n2) {

tab[i]=tab2[x2];

x2++;

}

}

fclose(plik1);

fclose(plik2);

free(tab1);

free(tab2);

return tab;

}

zapisz(char \*nazwa\_pliku, int \*tab, int n) {

FILE\* plik=fopen(nazwa\_pliku,"w");

if(!plik) {

printf("Nie udalo sie utworzyc pliku!");

exit(0);

}

int i;

for(i=0;i<n;i++)

fprintf(plik,"%d\n",tab[i]);

fclose(plik);

}

main() {

int \*tab, n, i;

char \*nazwa\_pliku1=(char\*)calloc(50, sizeof(char\*));

char \*nazwa\_pliku2=(char\*)calloc(50, sizeof(char\*));

char \*nazwa\_pliku3=(char\*)calloc(50, sizeof(char\*));

printf("Podaj nazwe pliku1 do wczytania wartosci: ");

gets(nazwa\_pliku1);

printf("Podaj nazwe pliku2 do wczytania wartosci: ");

gets(nazwa\_pliku2);

printf("Podaj nazwe pliku do zapisania wartosci: ");

gets(nazwa\_pliku3);

tab=wczytaj(nazwa\_pliku1, nazwa\_pliku2, tab, &n);

printf("\nPosortowane liczby:\n");

for(i=0;i<n;i++) {

printf("%d\t",tab[i]);

if((i+1)%10==0)

printf("\n");

}

zapisz(nazwa\_pliku3, tab, n);

free(nazwa\_pliku1);

free(nazwa\_pliku2);

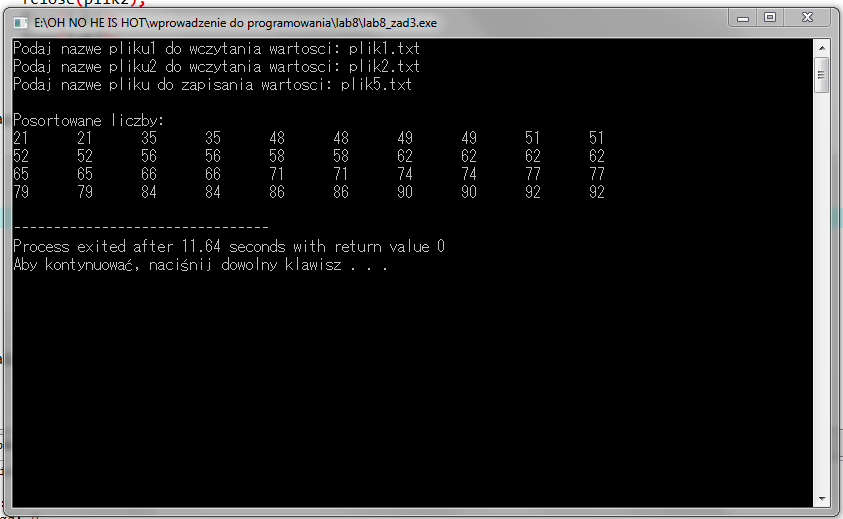
free(nazwa\_pliku3);

free(tab);

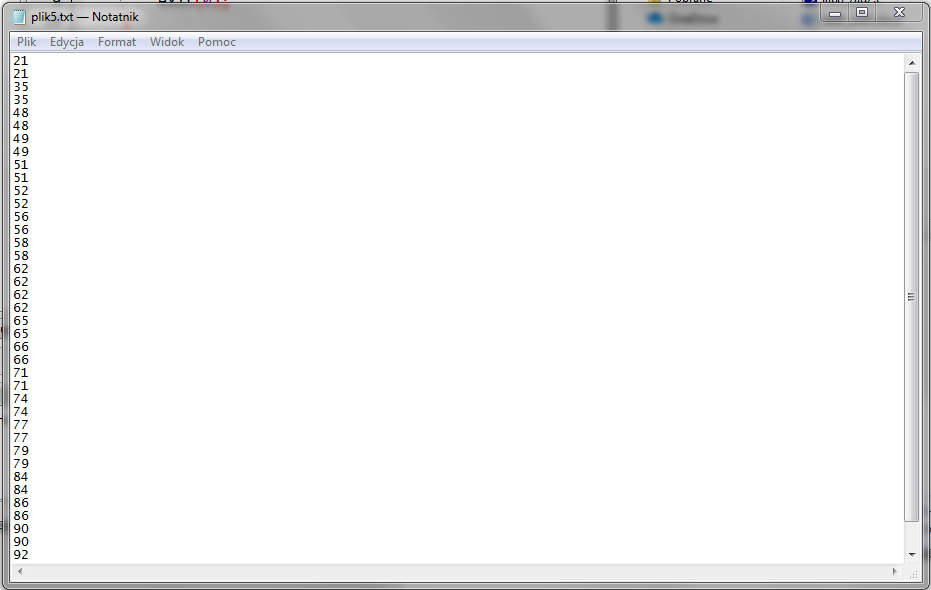
return 0;

}

* 1. Dane wejściowe



* 1. Dane wyjściowe



1. Wnioski

Złożoność obliczeniowa algorytmu:

O(n) = 2 + n1 + n2 + 1 + n1 + n2 + n + n + 1 = 2n + 2n1 + 2n2 + 4 = 2n + 2n + 4 = 4n + 4