



SPRAWOZDANIE

Zadanie projektowe nr. 1

Jakub Frydlewicz 169776

Inżynieria i analiza danych FSDI, grupa P02

1 OPIS PROBLEMU

1.1 TREŚĆ ZADANIA

Zadanie 5. Dla zadanej tablicy liczb całkowitych znajdź maksymalny iloczyn dwóch elementów znajdujących się w tablicy

Przykład.

Wejście $A[] = [-10, 5, 8, -4, 1]$

Wyjście: Czynniki generujące maksymalny iloczyn to pary: $[-10, -4]$ oraz $[5, 8]$

1.2 TABLICE DYNAMICZNE

Dane programów wykonywanych w komputerze są przechowywane w pamięci RAM (ang. Random Access Memory). Dotychczas napisane programy, choć w sposób ukryty, korzystały z niej podczas deklaracji funkcji, a później odwoływały się tam w celu odczytania lub modyfikacji zapisanych danych. Jednak programy nie wiedziałyby gdzie zapisać lub skąd odczytać jeśli nie będą wiedziały gdzie dane zostały zapisane.

Podczas uruchomienia napisanej aplikacji program deklaruje miejsce w pamięci pod adresy wykorzystywanych w programie zmiennych. W praktyce oznacza to, że w celu zadeklarowania zmiennej wysyła do systemu żądanie o adres wolnego miejsca w pamięci podając przy tym jak dużo pamięci potrzebuje i w wyniku tego otrzymuje adres. Odwołując się do tego adresu program może tam przechowywać dowolne informacje. Należy tu zauważyć, że wszystkie dane są przechowywane tak samo, ale to program decyduje jak je interpretować.

Na zakończenie działania program zwalania użyte zasoby poprzez wyrejestrowanie danego adresu z systemu. W ten sposób system wie, że daną część pamięci może przydzielić innemu programowi. Jednak w przypadku, gdy program zadeklaruje użycie pewnej części pamięci, ale jej nie zwolni to powstają tzw. wycieki pamięci. Z pozoru niegroźne, ponieważ po ponownym uruchomieniu komputera nie ma po nich śladu. Jednak powodują, że komputer ma mniej dostępnej pamięci, a to w rezultacie prowadzi do niestabilnej pracy całego systemu.¹

1.3 KRÓTKI OPIS DZIAŁANIA PROGRAMU

Po uruchomieniu program pyta o wybranie działania – wpisanie przez użytkownika liczb lub wygenerowanie liczb pseudolosowych. Sprawdza poprawność wprowadzonych danych. Jeśli dane są niepoprawne, wyświetla odpowiedni komunikat i pyta ponownie. Wprowadzone dane do tablicy są sortowane rosnąco i program sprawdza po kolei iloczyny sąsiednich liczb. W kolejnym etapie sprawdza największy iloczyn oraz wypisuje pary do pliku. Mierzy czas działania algorytmów. Wyniki programu zostają wprowadzone do pliku „output.txt” a następnie wyświetlone w programie.

¹ <https://mattomatti.com/pl/cp14>

2 PROGRAM

2.1 KOD

```
#include <iostream>
#include <algorithm>
#include <fstream>
#include <time.h>
#include <stdlib.h>

using namespace std;
clock_t start, stop;
double czas;
int ile = 0;
const string fileName = "output.txt";

int *stworzTablice()
{
    cout << "Podaj rozmiar tablicy: ";
    cin >> ile;
    while(cin.fail() || ile <= 1)
    {
        cout << "Niepoprawna wartosc: ";
        cin.clear();
        cin.ignore(100, '\n');
        cin >> ile;
    }

    int* tab = new int[ile];

    for(int i = 0; i < ile; i++)
    {
        cout << "Podaj element nr." << i+1 << ": ";
        cin >> tab[i];
        while( cin.fail() )
        {
            cout << "Wprowadz poprawna wartosc (liczbe calkowita): ";
            cin.clear();
            cin.ignore(100, '\n');
            cin >> tab[i];
        }
    }

    sort( tab, tab + ile );
    return tab;
}

void calcProduct( int tab[], int ile )
{
    ofstream file;
    file.open (fileName);

    int* Iloczyny = new int [ile - 1];
    // file << "\n\nIloczyny\n\n";

    for(int i=0; i<ile-1; i++)
    {
        Iloczyny[i] = tab[i] * tab[i+1];
        // file << i+1 << ". " << Iloczyny[i] << endl;
    }

    int maks = Iloczyny[0];
}
```

```

for(int j=0; j<ile - 2; j++)
{
    if( Iloczyny[j+1] > Iloczyny[j] )
    {
        maks = Iloczyny[j+1];
    }
}
file << "\nNajwiekszy iloraz: " << maks << endl;
file << "Czynniki generujace maksymalny iloczyn to pary: ";

for( int i=0; i< ile - 1; i++ )
{
    if( tab[i] * tab[i+1] == maks )
    {
        file << "[ " << tab[i] << ", " << tab[i+1] <<" ] ";
    }
}
file << "\n";
file.close();
}

int *randTab(){
cout<< "Wygenerowane liczby beda z zakresu (-99, 99)\n";
cout << "Podaj ile elementow ma zawierac tablica: ";
cin >> ile;
while( cin.fail() )
{
    cout << "Wprowadz poprawna wartosc (liczbe calkowita): ";
    cin.clear();
    cin.ignore(100, '\n');
    cin >> ile;
}
cout << "Trwa obliczenie..." << endl;
start = clock();
int *tab = new int[ile];
for(int i=0 ; i<ile ; i++){
    tab[i] = rand()%199-99;
}
sort(tab, tab + ile);
return tab;
}

void printFile()
{
    ifstream outcome(fileName);

    if ( outcome.is_open() )
    {
        cout << outcome.rdbuf();
        outcome.close();
    }
    else
    {
        cout << "Blad przy otwieraniu pliku: " << fileName;
    }
}

```

```

int main()
{
    cout << "1. Wprowadz dane" << endl;
    cout << "2. Wylosuj dane" << endl;
    int wybor;
    cin >> wybor;
    system("cls");
    srand(time(NULL));
    switch(wybor)
    {
        case 1:
        {
            int* tablica = stworzTablice();
            //for (int i=0; i<ile; i++)
            //  cout<<"1. " << tablica[i] << endl;

            calcProduct( tablica, ile );
            delete [] tablica;
            printFile();
            break;
        }

        case 2:
        {
            int *tabs = randTab();
            calcProduct ( tabs, ile);
            // for (int i=0; i<ile; i++)
            //  cout<<i+1<<" " << tabs[i] << endl;
            delete [] tabs;
            stop = clock();
            printFile();

            czas = (double)(stop - start) / CLOCKS_PER_SEC;
            cout << "\nCzas: " << czas << "s. ";
            break;
        }
        default:
            cout<<"Nieprawidlowy wybor!";
    }

    return 0;
}

```

Kod 1 Kod programu

2.2 ZASTOSOWANE OPERACJE

- Podanie rozmiaru tablicy dynamicznej
- Dodanie elementów tablicy dynamicznej
- Sprawdzenie poprawności wprowadzonych danych
- Sortowanie danych rosnąco
- Tworzenie pliku i zapisywanie w nim danych
- Odczyt pliku i wypisanie danych
- Wyszukanie największego iloczynu oraz wypisanie par generujących największy iloczyn
- Generowanie liczb losowych
- Mierzenie czasu działania algorytmu

2.3 MENU GŁÓWNE

1. Wprowadz dane
2. Wylosuj dane

2.4 PRZYKŁADOWE DZIAŁANIE PROGRAMU

Podaj rozmiar tablicy: 5

Podaj element nr.1: -10

Podaj element nr.2: 5

Podaj element nr.3: 8

Podaj element nr.4: -4

Podaj element nr.5: 1

Największy iloraz: 40

Czynniki generujące maksymalny iloczyn to pary: [-10, -4] [5, 8]

1 Przykładowe działanie programu dla wpisanych liczb przez użytkownika

2.5 TESTOWE DZIAŁANIE PROGRAMU

Podaj rozmiar tablicy: 8

Podaj element nr.1: -8

Podaj element nr.2: 4

Podaj element nr.3: 9

Podaj element nr.4: -9

Podaj element nr.5: -8

Podaj element nr.6: 0

Podaj element nr.7: 9

Podaj element nr.8: 3

Największy iloraz: 81

Czynniki generujące maksymalny iloczyn to pary: [9, 9]

2 Testowe działanie programu dla liczb wprowadzonych przez użytkownika

Podaj rozmiar tablicy: 6

Podaj element nr.1: -5

Podaj element nr.2: 10

Podaj element nr.3: -10

Podaj element nr.4: 5

Podaj element nr.5: 5

Podaj element nr.6: -5

Największy iloraz: 50

Czynniki generujące maksymalny iloczyn to pary: [-10, -5] [5, 10]

3 Testowe działanie programu dla liczb wprowadzonych dla użytkownika

Wygenerowane liczby beda z zakresu (-99, 99)

Podaj ile elementow ma zawierac tablica: 15

Trwa obliczenie...

Najwiekszy iloraz: 8526

Czynniki generujace maksymalny iloczyn to pary: [87, 98]

4 Testowe dzialanie programu dla liczb pseudolosowych

Wygenerowane liczby beda z zakresu (-99, 99)

Podaj ile elementow ma zawierac tablica: 10

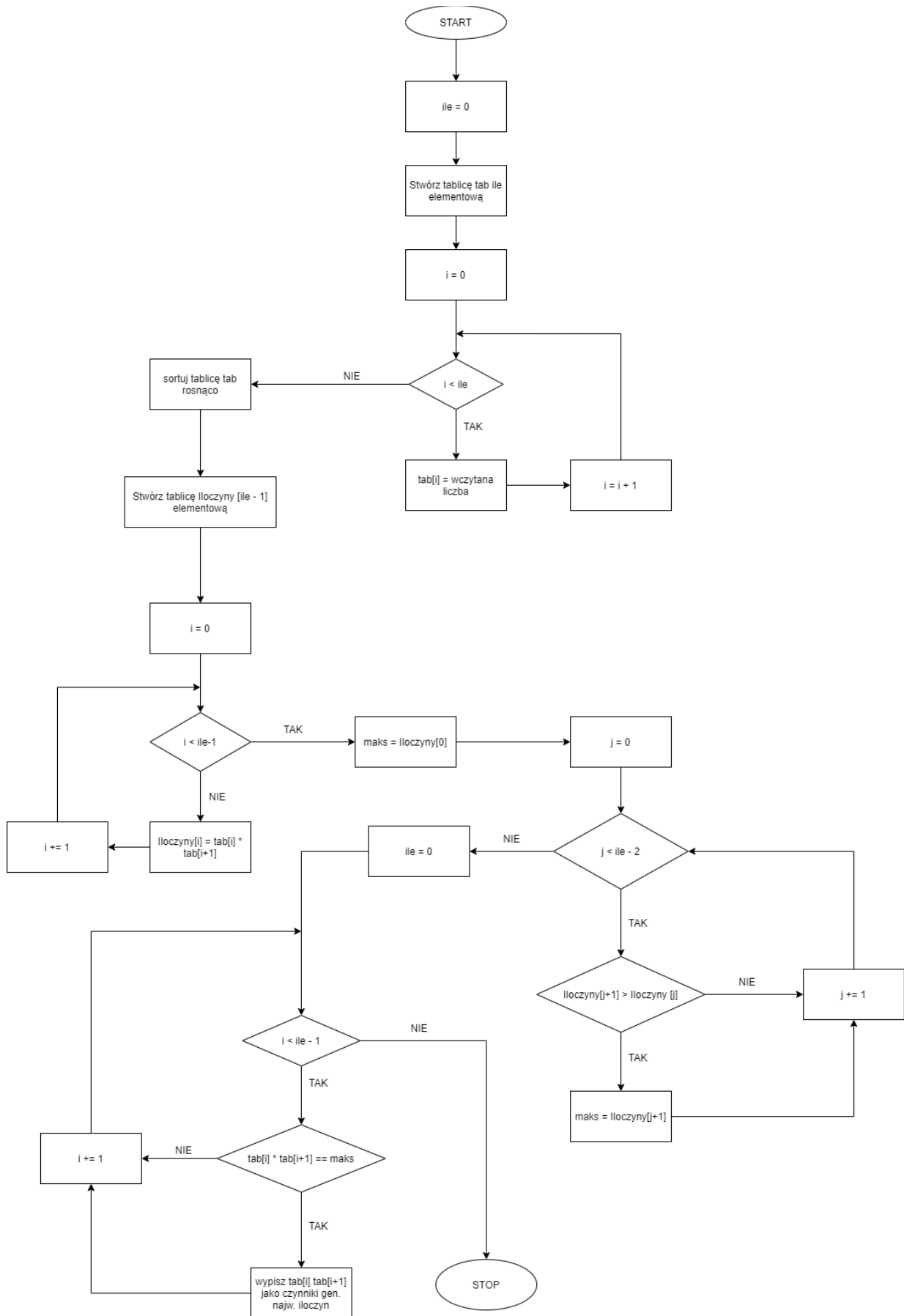
Trwa obliczenie...

Najwiekszy iloraz: 5742

Czynniki generujace maksymalny iloczyn to pary: [66, 87]

5 Testowe dzialanie programu dla liczb pseudolosowych

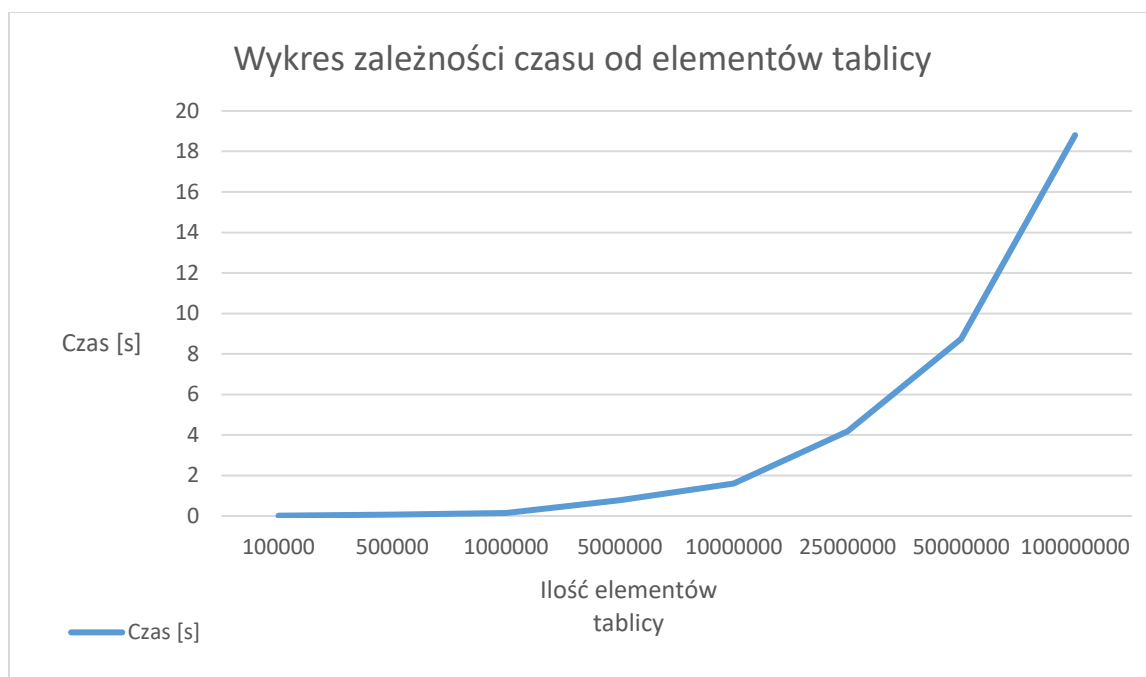
2.6 SCHEMAT BLOKOWY



2.7 PSEUDOKOD

```
ile ← 0
ile ← rozmiar tablicy
Stwórz tablice tab [ile] elementową
Dla ( i = 0; i < ile; i++ ):
    tab[i] ← wprowadz element
Posortuj wszystkie elementy tablicy tab rosnąco
Zwróć tablicę tab
Stwórz tablicę Iloczynny [ile - 1] elementową
Dla ( i=0; i < ile-1; i++):
    Iloczynny[i] = tab[i] * tab[i+1]
maks ← Iloczynny[0]
Dla ( j=0; j<ile-2; j ++ ):
    Jeżeli ( Iloczynny[j+1] > Iloczynny[j] ):
        Maks = Iloczynny[j+1]
Dla ( i=0; i < ile - 1; i ← i - 1 ):
    Jeżeli ( tab[i] * tab[i+1] == maks):
        Wypisz oba czynniki generujące największy iloczyn(tab[i] oraz
        tab[i+1])
```

2.8 ZŁOŻONOŚĆ CZASOWA



3 DOKUMENTACJA Z DOŚWIADCZEŃ

3.1 PROGRAM WYŚWIETLA BŁĄD PODCZAS WPROWADZENIA NIEPOPRAWNEGO ROZMIARU TABLICY LUB ELEMENTU

Podaj rozmiar tablicy: j

Niepoprawna wartosc: 2

Podaj element nr.1: i

Wprowadz poprawna wartosc (liczbe calkowita): 1

Podaj element nr.2: 1

Najwiekszy iloraz: 1

Czynniki generujace maksymalny iloczyn to pary: [1, 1]

6 Wynik wprowadzenia niepoprawnych danych

4 WNIOSKI I PODSUMOWANIE

Program za pomocą funkcji, pętli, instrukcji warunkowych pozwala na wykonanie zadania. Sprawdza poprawność wprowadzanych danych, w przypadku wprowadzenia błędnych danych wypisuje błąd i pozwala na ponowne wprowadzenie danych. Pozwala na tworzenie tablic dynamicznych i generowanie liczb pseudolosowych. Mierzy czas operacji.

5 SPIS TREŚCI

1	Opis problemu.....	1
1.1	Treść zadania	1
1.2	Tablice dynamiczne	1
1.3	Krótki opis działania programu	1
2	Program.....	2
2.1	Kod	2
2.2	Zastosowane operacje.....	5
2.3	Menu Główne	5
2.4	Przykładowe działanie programu	5
	Testowe działanie programu.....	6
2.5	Schemat Blokowy	8
2.6	Pseudokod.....	9
2.7	Złożoność czasowa	10
3	Dokumentacja z doświadczeń.....	10
3.1	Program wyświetla błąd podczas wprowadzenia niepoprawnego rozmiaru tablicy lub elementu.....	10
4	Wnioski i podsumowanie.....	11