



POLITECHNIKA RZESZOWSKA im.
Ignacego Łukasiewicza
WYDZIAŁ MATEMATYKI I FIZYKI STOSOWANEJ

ANDRII KOTOVYCH
173163

ALGORYTMY I STRUKTURY DANYCH

Projekt

kierunek studiów: Inżynieria i analiza danych

Opiekun pracy:

Prof. Mariusz Borkowski

Rzeszów 2022

Spis treści

1. Wstęp.....	3
2. Algorytm.....	3
3. Schemat blokowy.....	4
4. Pseudokod.....	5
5. Rezultaty testów.....	5
6. Wykresy złożoności czasowej oraz obliczeniowej.....	7
7. Wnioski oraz podsumowanie.....	7
8. Kod	8

1. Wstęp

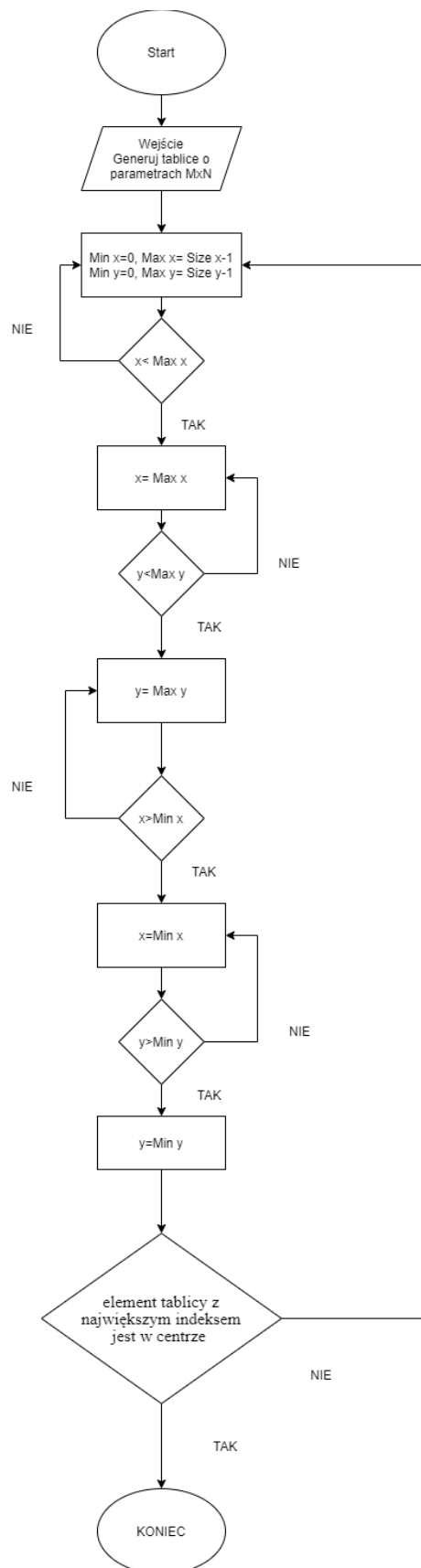
Zagadnienie projektu polega na wypisaniu elementów tablicy dwuwymiarowej o rozmiarze $M \times N$ „obchodząc ją po spirali”.

2. Algorytm

Algorytm opiera się na zwiększaniu indeksa jednej z tablic (odczytać idąc w prawo), zwiększaniu indeksa drugiej z tablic (odczytać idąc w dół), zmniejszaniu indeksa pierwszej tablicy (odczytać idąc w lewo), zmniejszaniu indeksa drugiej tablicy (odczytać idąc w górę).

3. Schemat blokowy

Rysunek 1 - Schemat blokowy.



Rysunek 1 Schemat blokowy

4. Pseudokod

K01: Generuj tablice dwówymiarową o parametrach $M \times N$

K02: Początkowe parametry x i y

K03: $x = \text{Min } x$, dopóki $x < \text{Max } x$ wykonuj, po wykonaniu $\text{Max } x$ zmniejsza się o 1 element

K04: $y = \text{Min } y$, dopóki $y < \text{Max } y$ wykonuj, po wykonaniu $\text{Max } y$ zmniejsza się o 1 element

K05: $x = \text{Max } x$, dopóki $x > \text{Min } x$ wykonuj, po wykonaniu $\text{Min } x$ zmniejsza się o 1 element

K06: $y = \text{Max } x$, dopóki $y > \text{Min } y$ wykonuj, po wykonaniu $\text{Min } y$ zmniejsza się o 1 element

K07: Jeżeli element tablicy z największym indeksem jest w centrum skończ działanie,

Jeżeli nie powtórz K03-K06

5. Rezultaty testów

1	2	3																																	
8	9	4																																	
7	6	5																																	
1	2	3	4	5	6	7	8	9																											
1	2	3	4																																
12	13	14	5																																
11	16	15	6																																
10	9	8	7																																
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16																				
1	2	3	4	5																															
16	17	18	19	6																															
15	24	25	20	7																															
14	23	22	21	8																															
13	12	11	10	9																															
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25											
1	2	3	4	5	6																														
20	21	22	23	24	7																														
19	32	33	34	25	8																														
18	31	36	35	26	9																														
17	30	29	28	27	10																														
16	15	14	13	12	11																														
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36

```

1 2 3 4 5 6 7
24 25 26 27 28 29 8
23 40 41 42 43 30 9
22 39 48 49 44 31 10
21 38 47 46 45 32 11
20 37 36 35 34 33 12
19 18 17 16 15 14 13

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43
44 45 46 47 48 49

1 2 3 4 5 6 7 8
28 29 30 31 32 33 34 9
27 48 49 50 51 52 35 10
26 47 60 61 62 53 36 11
25 46 59 64 63 54 37 12
24 45 58 57 56 55 38 13
23 44 43 42 41 40 39 14
22 21 20 19 18 17 16 15

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43
44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64

1 2 3 4 5 6 7 8 9
32 33 34 35 36 37 38 39 10
31 56 57 58 59 60 61 40 11
30 55 72 73 74 75 62 41 12
29 54 71 80 81 76 63 42 13
28 53 70 79 78 77 64 43 14
27 52 69 68 67 66 65 44 15
26 51 50 49 48 47 46 45 16
25 24 23 22 21 20 19 18 17

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43
44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81
Process finished with exit code 0

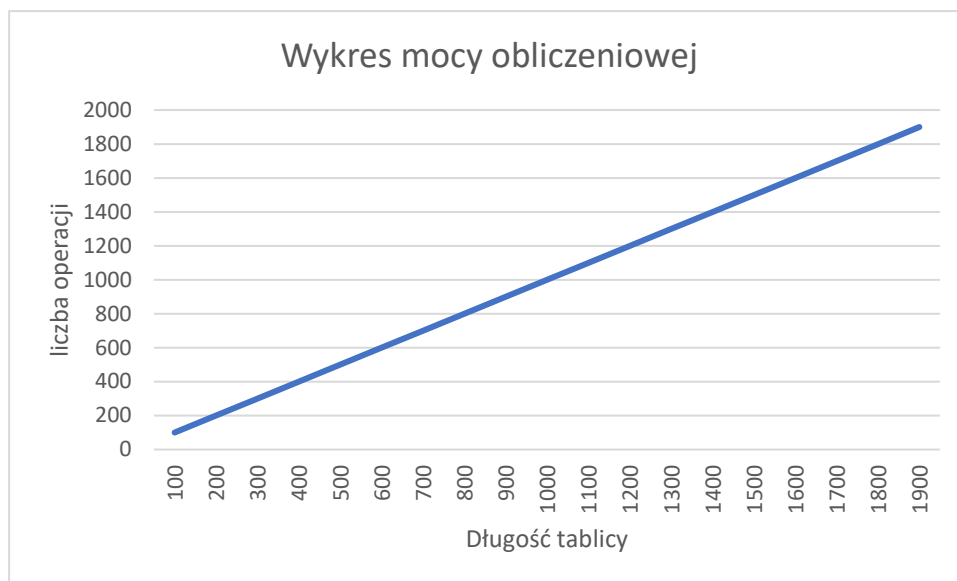
1 2 3 4 5 6 7 8 9 10
36 37 38 39 40 41 42 43 44 11
35 64 65 66 67 68 69 70 45 12
34 63 84 85 86 87 88 71 46 13
33 62 83 96 97 98 89 72 47 14
32 61 82 95 100 99 90 73 48 15
31 60 81 94 93 92 91 74 49 16
30 59 80 79 78 77 76 75 50 17
29 58 57 56 55 54 53 52 51 18
28 27 26 25 24 23 22 21 20 19

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43
44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83
84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

```

6. Wykresy złożoności czasowej oraz obliczeniowej

Moc obliczeniowa 3Ghz (około 3000000000 operacji/sekunda)



7. Wnioski oraz podsumowanie

Algorytm szybki i uniwersalny, działa dla każdej tablicy dwuwymiarowej o dowolnych parametrach $M \times N$.

Samodzielnie generuje tablice i nie potrzebuje pobrania danych z plików.

8. Kod

```
#include <iostream>
#include <iomanip>
using namespace std;

int main()
{
    // Pobieranie rozmiarów tablicy podanych przez użytkownika
    const int SizeX=5,SizeY=5;
    // Tworzenie tabeli
    int tab[SizeY][SizeX];
    int MinX=0,MaxX=SizeX-1,MinY=0,MaxY=SizeY-1;
    // Deklaracja zmiennej "żyjącej" tylko na czas nieskończonej pętli
    for(int V=0;;)
    {
        // Algorytm wpisywania liczb w tablicę
        for(int x=MinX;x<=MaxX;++x) tab[MinY][x]=++V;
        if((0>MaxX) || (++MinY>MaxY)) break;
        for(int y=MinY;y<=MaxY;++y) tab[y][MaxX]=++V;
        if((0>--MaxX) || (MinY>MaxY)) break;
        for(int x=MaxX;x>=MinX;--x) tab[MaxY][x]=++V;
        if((0>MaxX) || (MinY>--MaxY)) break;
        for(int y=MaxY;y>=MinY;--y) tab[y][MinX]=++V;
        if(++MinX>MaxX || (MinY>MaxY)) break;
    }
    for(int y=0;y<SizeY;++y,cout<<endl) for(int x=0;x<SizeX;++x)
cout<<setw(4)<<tab[y][x];

    cout << "\n" << endl;

    {
        // Pobiera podane rozmiary
        for(int i=1;i<=SizeX*SizeY;i++){
            // Wypisuje wartości tabeli od 1
            std::cout<<i<<" ";
        }
    }

    return 0;
}
```