

Rzeszów, 15.11.2021

Sprawozdanie  
z projektu

Politechnika Rzeszowska  
Algorytmy i struktury danych

Krzysztof Motas  
Inżynieria i analiza danych, 1 rok, grupa nr 5

## Spis treści

|    |                                 |   |
|----|---------------------------------|---|
| 1. | Opis problemu .....             | 3 |
| 2. | Opis algorytmu .....            | 3 |
| 3. | Pseudokod .....                 | 3 |
| 4. | Schemat blokowy .....           | 4 |
| 5. | Testy działania algorytmu.....  | 5 |
| 6. | Wykres złożoności czasowej..... | 6 |

## 1. Opis problemu

Dla zadanej tablicy liczb całkowitych znajdź te pary, których różnica jest równa zadanej liczbie  $k$ .

## 2. Opis algorytmu

Rozpoczynamy od wprowadzenia do funkcji `FindPairOfNumbers` wektora przechowującego liczby całkowite oraz poszukiwanej wartości różnicy. Tworzymy mapę, w której będą przechowywane znalezione pary. Uruchamiamy dwie pętle za pomocą instrukcji iteracyjnej `for`, o zakresie długości liczby elementów podanego wektora. Porównujemy za pomocą instrukcji `if` wartość bezwzględną liczby z pierwszej pętli z liczbą z drugiej pętli. Dodatkowo musimy sprawdzić, czy różnica drugiej liczby i pierwszej liczby nie równa się szukanej wartości różnicy – ma to na celu wykluczenie powtarzających się przypadków, np.  $[2,5]$ ,  $[5,2]$ . Jeśli para liczb spełnia wyżej wymienione warunki, dodajemy ją do stworzonej mapy. Funkcja kończy się zwracaniem mapy, zawierającej znalezione pary liczb.

## 3. Pseudokod

```
wczytaj(array)
wczytaj(diff)

foundPairs
i <- 0
j <- 0
size <- rozmiar tablicy array

dopóki j < size wykonuj
  i <- array[j]
  c <- 0
  x <- 0

  dopóki c < size wykonuj
    x <- array[c]

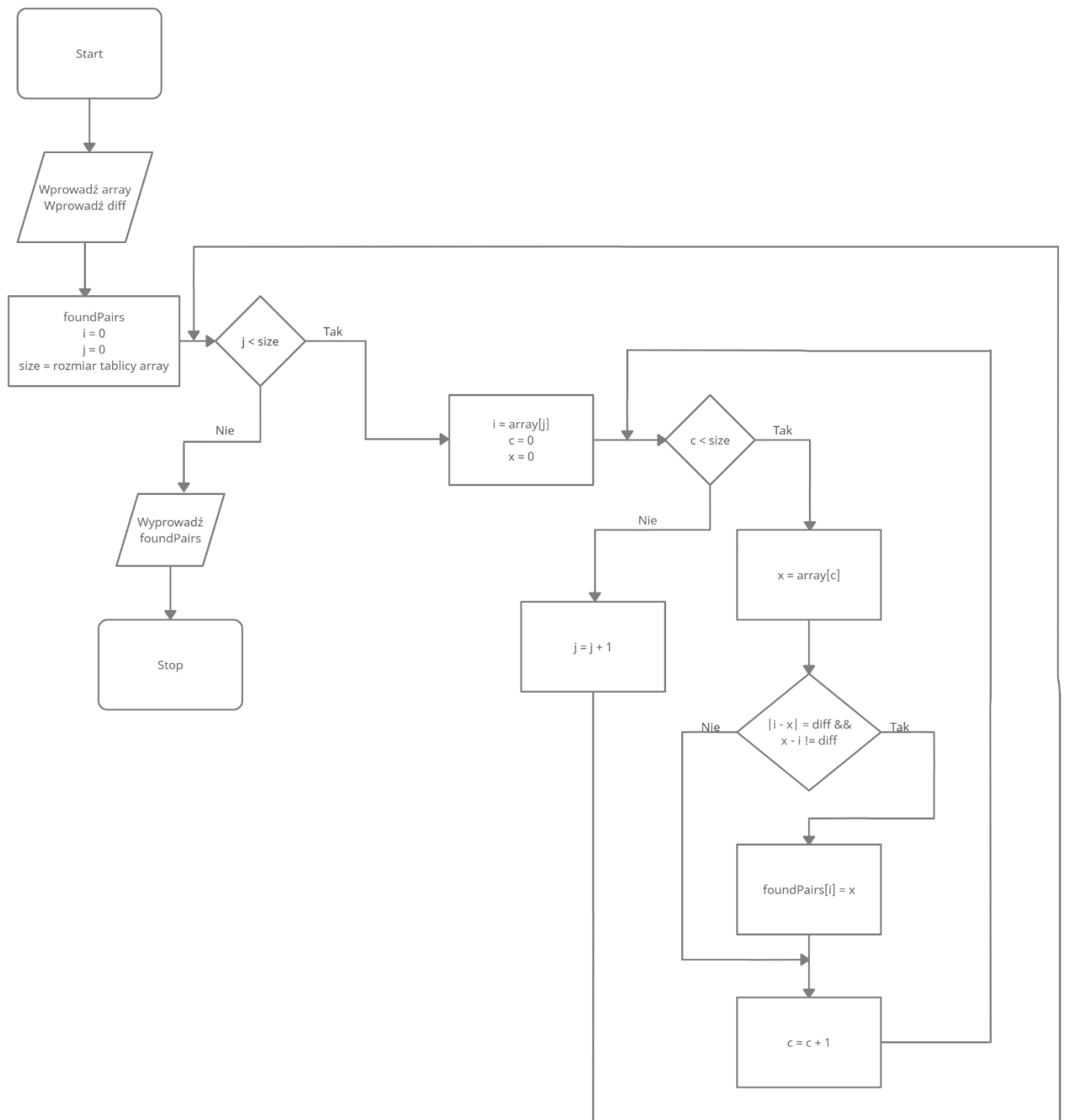
    jeśli  $|i - x| = \text{diff}$  oraz  $x - i \neq \text{diff}$  to
      foundPairs[i] <- x

    c <- c + 1

  j <- j + 1

zwróć foundPairs
```

#### 4. Schemat blokowy



## 5. Testy działania algorytmu

- a) Liczba elementów w tablicy: 30, szukana różnica: 8

Dane wejściowe: 6 14 5 15 7 2 12 14 15 7 1 2 9 5 13 1 8 5 10 9 4 6 4 8 2 10 5 4 2 10

Dane wyjściowe: [1,9] [2,10] [4,12] [5,13] [6,14] [7,15]

- b) Liczba elementów w tablicy: 20, szukana różnica: 12

Dane wejściowe: 6 8 8 17 8 2 4 3 11 7 13 17 12 4 9 10 8 7 3 13

Dane wyjściowe: Nie znaleziono żadnych par.

- c) Liczba elementów w tablicy: 100, szukana różnica: 34

Dane wejściowe: 43 13 34 88 22 65 27 43 61 69 32 43 81 18 35 22 32 22 18 22 66 24 87 65 74  
64 78 39 72 21 14 55 47 68 16 57 25 45 82 9 57 38 51 69 87 15 77 76 14 39 44 30 71 71 64 13  
43 22 59 61 21 23 32 30 60 52 33 55 39 36 43 73 70 43 62 38 24 66 80 77 83 82 52 21 47 89  
69 71 40 68 85 64 79 31 72 33 76 72 10 46

Dane wyjściowe: [9,43] [10,44] [13,47] [18,52] [21,55] [23,57] [25,59] [27,61] [30,64] [31,65]  
[32,66] [34,68] [35,69] [36,70] [38,72] [39,73] [40,74] [43,77] [44,78] [45,79] [46,80] [47,81]  
[51,85] [55,89]

- d) Liczba elementów w tablicy: 150, szukana różnica: 140

Dane wejściowe: 114 21 113 67 103 36 127 120 73 79 13 144 138 2 10 62 80 136 147 135 82  
5 89 105 88 50 52 93 89 117 98 50 89 44 21 106 100 98 81 38 116 105 42 28 75 55 135 132  
95 130 58 83 128 2 143 63 20 131 60 93 79 145 38 122 101 89 117 104 4 95 14 20 68 117 69  
95 77 143 76 24 106 119 52 139 29 28 18 5 95 50 17 29 121 15 56 38 24 6 95 122 142 78 17  
103 4 128 48 123 81 143 136 59 137 97 21 130 22 57 17 96 98 13 49 73 142 61 145 20 139 19  
87 141 99 84 55 112 142 52 89 10 135 108 24 107 87 90 53 33 46 85

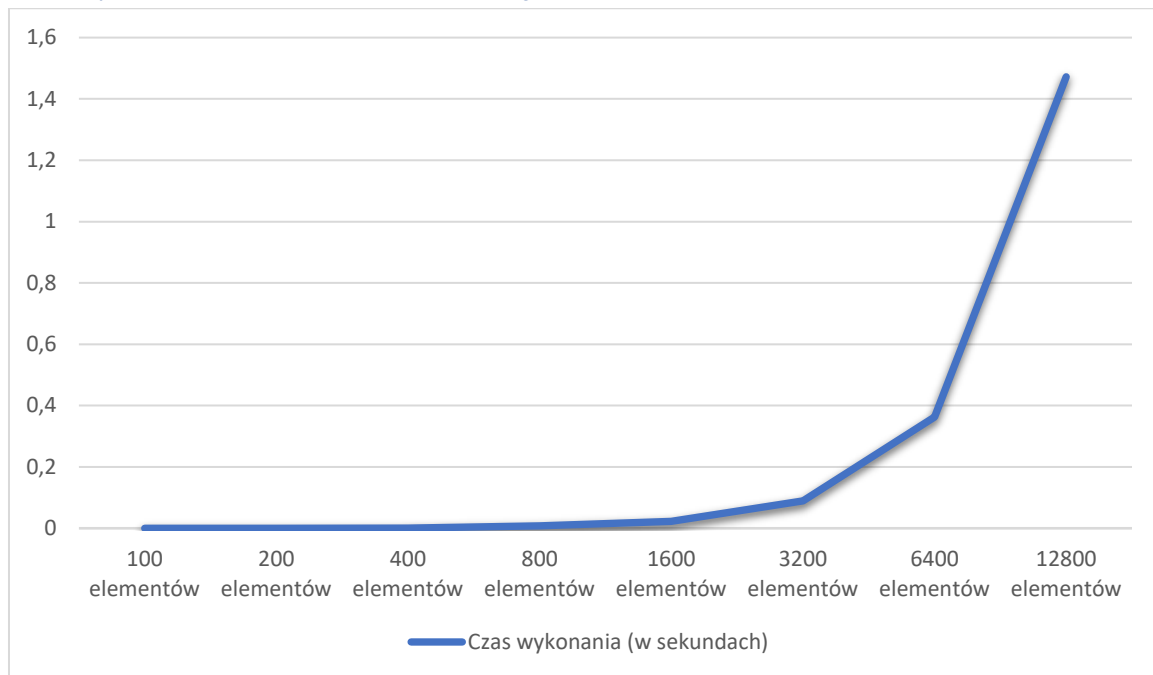
Dane wyjściowe: [2,142] [4,144] [5,145]

- e) Liczba elementów w tablicy: 15, szukana różnica: 3

Dane wejściowe: -20 -9 7 -7 -4 5 3 0 -12 10 13 -6 4 18 -1

Dane wyjściowe: [-12,-9] [-9,-6] [-7,-4] [-4,-1] [0,3] [4,7] [7,10] [10,13]

## 6. Wykres złożoności czasowej



Wykres skonstruowany na podstawie wygenerowanych danych z funkcji:

```
void DoTimeMeasurements()
{
    clock_t time;

    fstream file;
    file.open("measurements.txt", ios::out);

    for (int i = 1, elements = 100; i < 9; i++)
    {
        time = clock();
        FindPairOfNumbers(GenerateRandomNumbers(1, 500, elements),
10);

        file << "Czas wykonania funkcji \"FindPairOfNumbers\" dla "
<< elements << " elementów wynosi ~" << (float)(clock() - time) /
CLOCKS_PER_SEC << " s.\n";
        elements *= 2;
    }

    file.close();
}
```