Politechnika Rzeszowska, 27.11.2020r.

# Zadanie projektowe nr 1 Algorytmy i Struktury Danych

# Inżynieria i analiza danych, I rok

Kamil Hansel

Numer indeksu: 166728

Grupa: P03

## Opis problemu

### Treść zadania

Dla ciągu (w postaci tablicy) zawierającego wyłącznie wartości 0 lub 1, znajdź podciąg zawierający równą liczbę zer i jedynek, którego długość jest największa.

**Przykład:**

Wejście: 0, 0, 1, 0, 1, 0, 0

Wyjście: Najdłuższym podciągiem są 0, 1, 0, 1 oraz 1, 0, 1, 0

### Przeznaczenie programu

Program ma za zadanie znaleźć w ciągu (w postaci tablicy) zawierającego wyłącznie wartości 0 lub 1, podciąg o możliwie największej długości.

### Tablice w C++

[[1]](#footnote-1) **Tablica** ( ang. array ) lub **wektor** ( ang. vector ) jest **złożoną strukturą danych** ( ang. compound data structure ) zbudowaną z ciągu elementów tego samego typu. W pamięci komputera elementy tablicy są ułożone kolejno jeden obok drugiego. Dostęp do elementu odbywa się poprzez numer zwany indeksem. Na podstawie indeksu, rozmiaru elementu oraz adresu początku tablicy komputer oblicza adres elementu i w ten sposób uzyskujemy do niego dostęp.

We współczesnych językach programowania tablice są stosowane powszechnie do przechowywania danych podobnego rodzaju. Przy ich pomocy można zapisywać ciągi liczbowe, wyniki pomiarów różnych wielkości oraz tworzyć złożone bazy danych. Liczba zastosowań tablic jest w zasadzie ograniczona naszą wyobraźnią. Podstawową zaletą tablic jest prostota przetwarzania ich elementów. Dzięki dostępowi poprzez indeksy, elementy tablic daje się łatwo przetwarzać w pętlach iteracyjnych.

W tym rozdziale zajmujemy się algorytmami wyszukiwania danych w tablicach zawierających liczby. Jednakże podane tutaj algorytmy można z powodzeniem uogólnić na dane dowolnego typu, dlatego istotne jest zrozumienie sposobu pracy opisanych algorytmów

Rozróżnia się dwa typy tablic: statyczne - czyli takie, których rozmiaru podczas działania programu nie da się zmienić; oraz dynamiczne - czyli takie, którym pamięć jest przypisywana w trakcie działania programu.

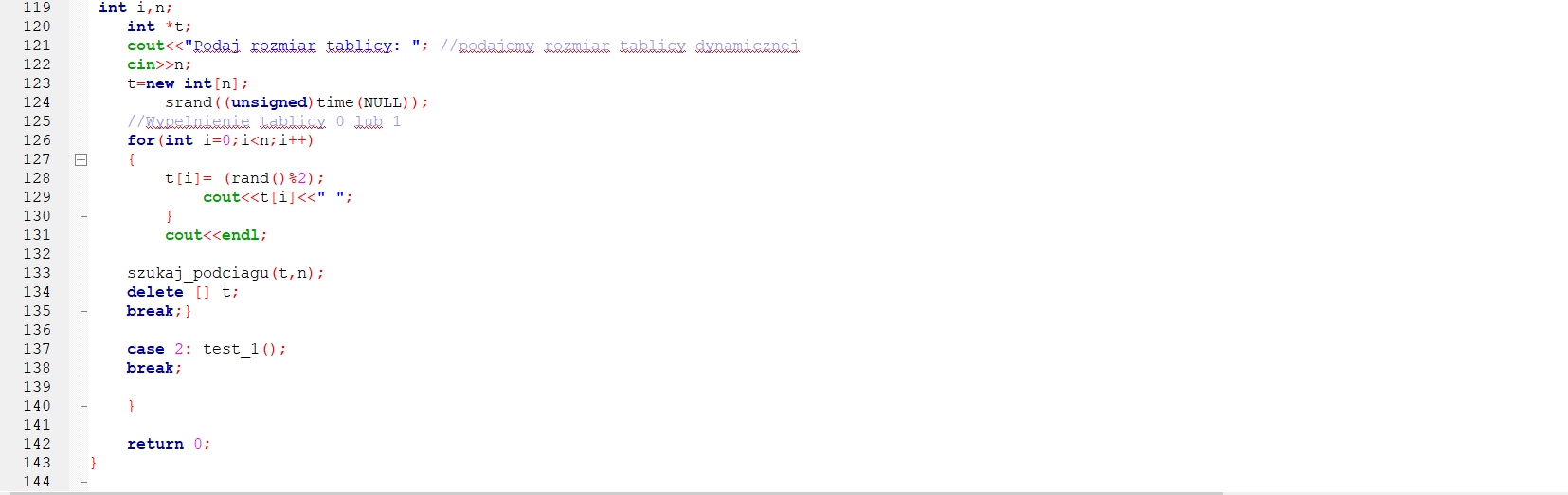
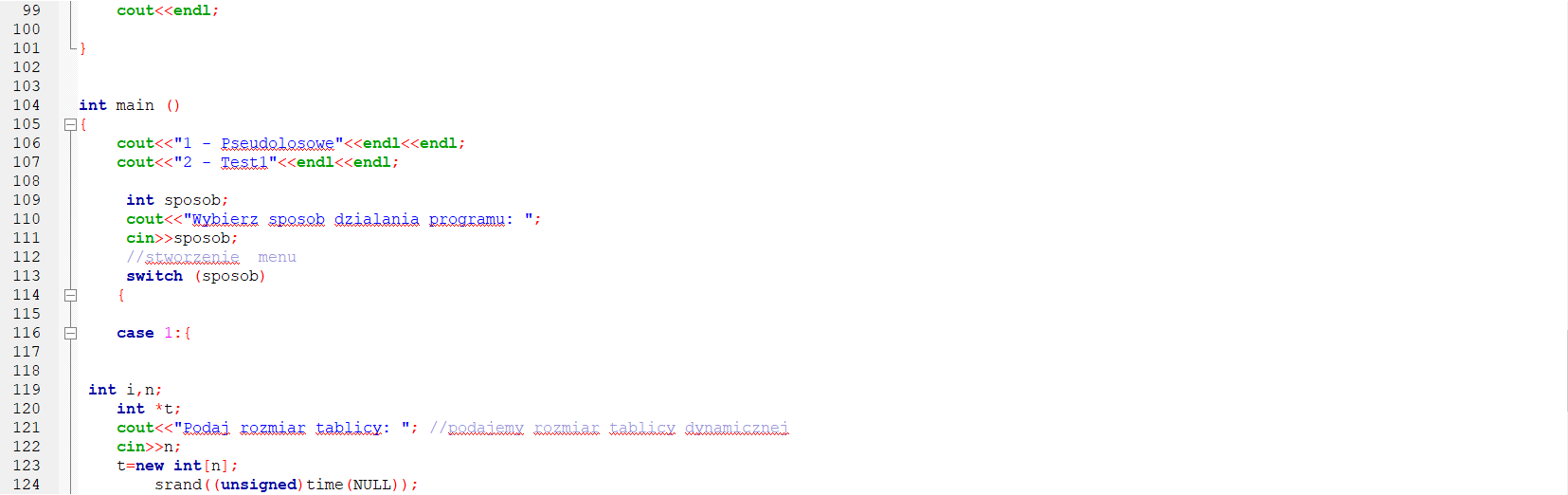
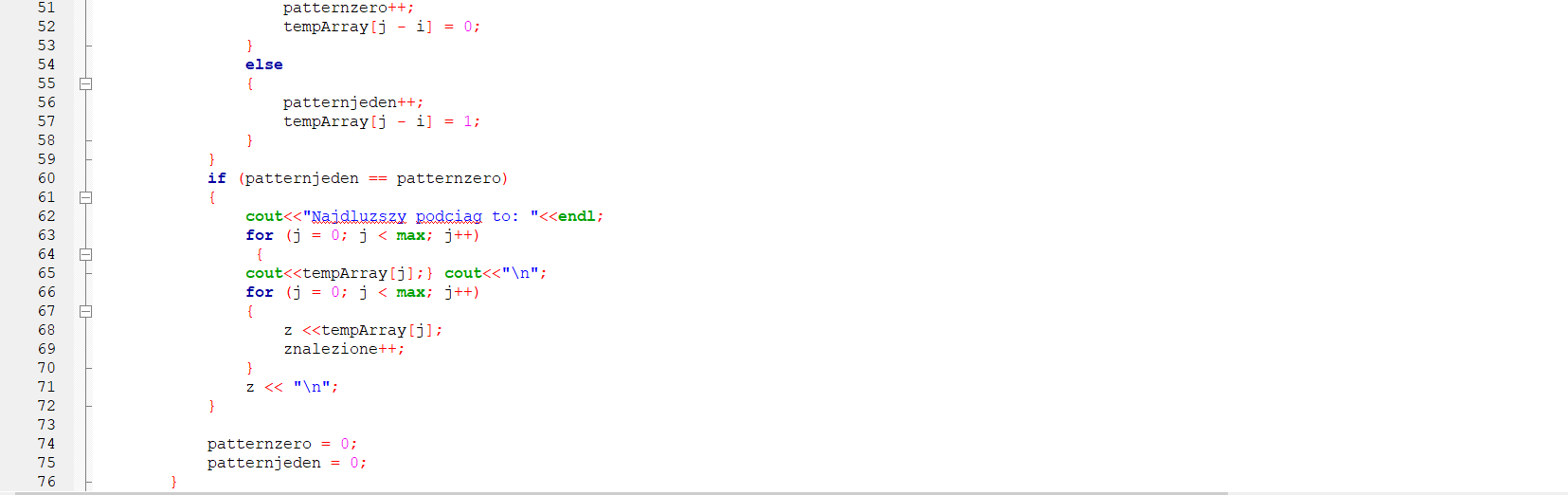
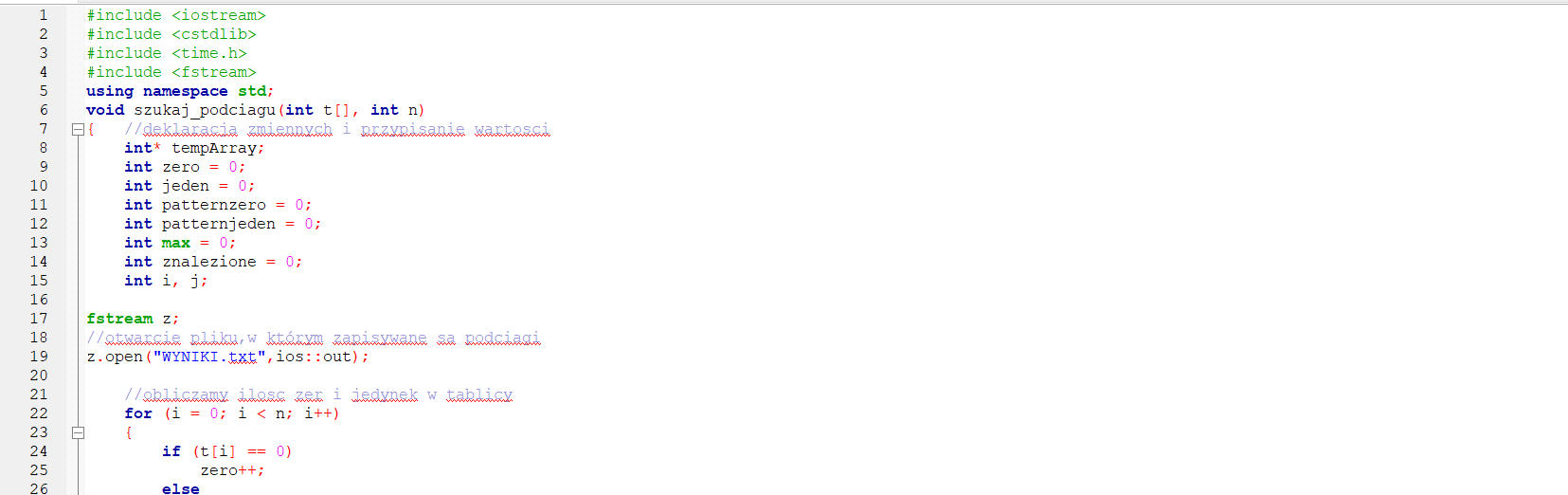
### Krótka charakterystyka programu

Program ma na celu odnalezienie najdłuższego podciągu dla ciągu (w postaci tablicy) zawierającego wyłącznie wartości 0 i 1. Na początku definiujemy naszą tablicę wejściową.

Po wykonaniu tego zadania, obliczamy ilość zer i jedynek w tablicy. Następnie wyliczamy maksymalną długość ciągu. W późniejszych krokach przeszukujemy tablicę wejściową.   
Po uruchomieniu programu na konsoli wyświetlają się wyniki oraz stwarza nam się plik tekstowy o nazwie „rozwiązania.txt”, w którym zapisują się najdłuższe podciągi.

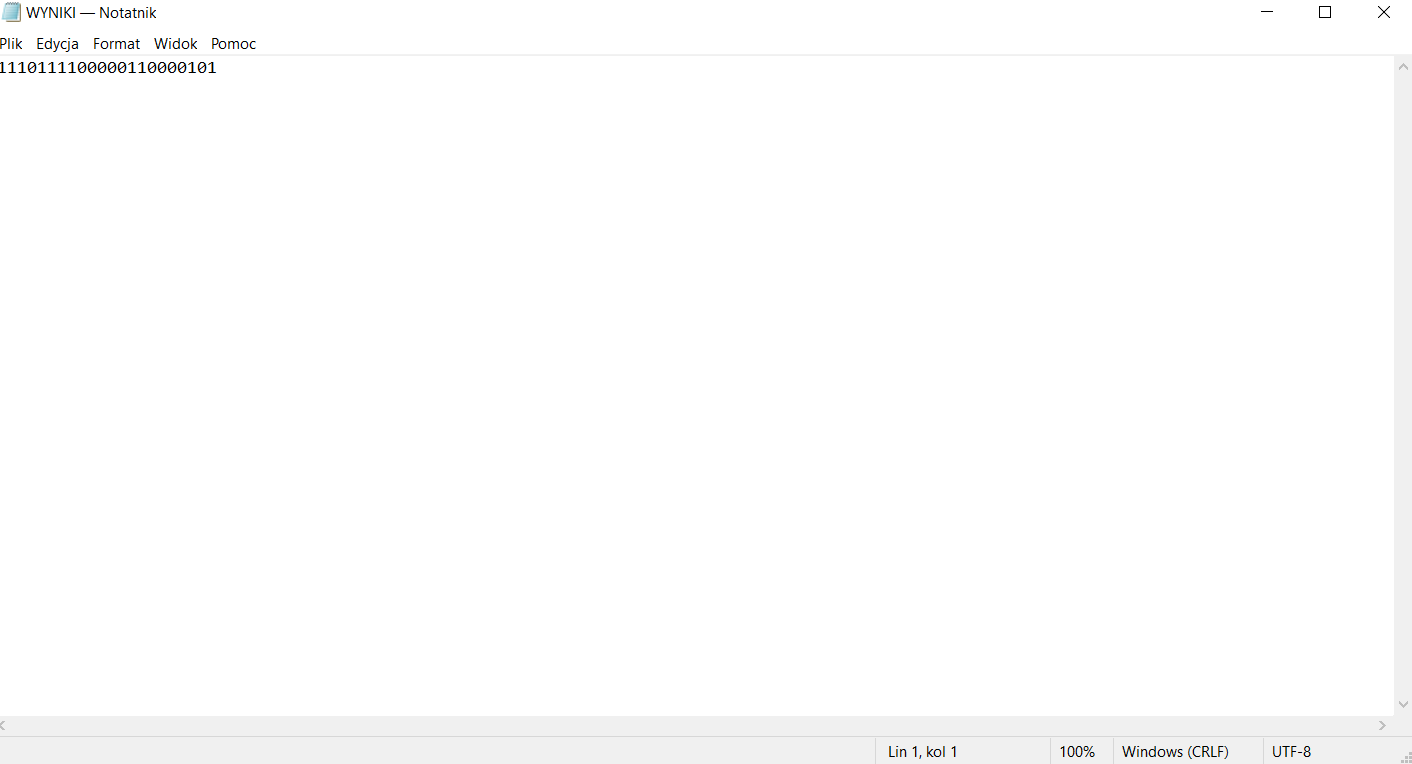
## Rozwiązanie zadania:

### Kod programu



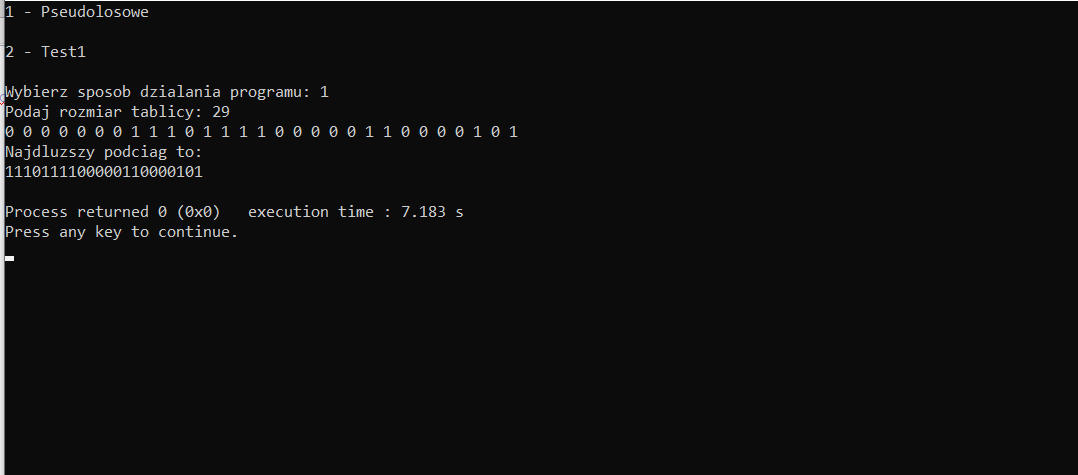
**Program wykonany w środowisku Code::Blocks IDE w języku C++**

### Rozwiązania wyświetlone w pliku tekstowym



Rysunek 1 Wynik wyświetlony w pliku tekstowym

### Rozwiązania wyświetlone na konsoli



Rysunek Wynik wyświetlony na konsoli

### Pseudokod

\* tymczasowyCiąg

zero←0

jeden←0

patternzero←0

patternjeden←0

maxdlugosc←0

znajdzCiąg←0

i,j

fstream z

Otwórz plik 1 o nazwie „WYNIKI.txt”

Dla i←0;i<n; i++ wykonuj

Jeżeli t[i]←0 to

zero zwiększamy o 1

w innym przypadku

jeden zwiększamy o 1

Jeżeli zero<jeden to

max←zero\*2

Jeżeli jeden<zero to

max←jeden\*2

Jeżeli jeden==zero to

max←jeden\*2

tymczasowyCiąg←nowy[max]

Kiedy max>0

Dla i←0;i<=n-max;i++ wykonuj

patternzero←0

patternjeden←0

Dla j←0+i;j<max+1 j++ wykonuj

jeżeli tab[j]←0 to

patternzero zwiększamy o 1

tymczasowyCiąg[j-i]←0

w przeciwnym razie

patternjeden zwiększamy o 1

tymczasowyCiąg[j-i]←1

Jeżeli (patternjeden==patternzero) to

Wypisz”Najdłuższy podciąg to:” zakończ linię

Dla j←0; j<max-; j++ wykonuj

Wypisz tymczasowyCiąg[j]

Wypisz”\n”

Dla(j←0;j<max;j++wykonuj

Wypisz”Najdłuższy podciąg to: ”zakończ linię

Dla j←0; j<max; j++ wykonuj

z <<tymczasowyciag[j]

Wypisz”n”

Dla j←0;j<max; j++ wykonuj

z <<tymczasowyCiąg[j]

Znalezione++;

z<<„\n”;

patternzero←0

patternjeden←0

Jeżeli znalezione>0 wykonuj

Usuń[]tymczasowyCiąg

Break

Max—

Jeżeli znalezione>0

Usuń[]tymczasowyCiąg

z.zamknij

Wykonuj

Wypisz”Test 1” zakończ linię zakończ linię

a[2]←{0,1}

Wypisz”A[]←„

Dla i←0;i<2;i++ Wypisz a[i] ” „

Wypisz zakończ linie

Szukaj\_podciągu(a,2)

Wypisz zakończ linie

Main() wykonuj

Wypisz”1 Pseudolosowe”zakończ linię zakończ linię

Wypisz”2 – Test1” zakończ linię zakończ linię

Sposób

Wypisz”Wybierz sposób działania programu: „

Wpisz sposób

Zmień sposób

Wykonaj

Przypadek 1 wykonaj

i,n

\*t

Wypisz”podaj rozmiar tablicy:

Wpisz n

T←new[n]

Srand(niezapisane)czas(NULL)

Dla i←0;i<n;i++wykonaj

T[i] ←(zaokrąglij()%2)

Wypisz t[i]<<” „

Wpisz zakończ linie

Szukaj podciągu(t,n)

Usuń[]t

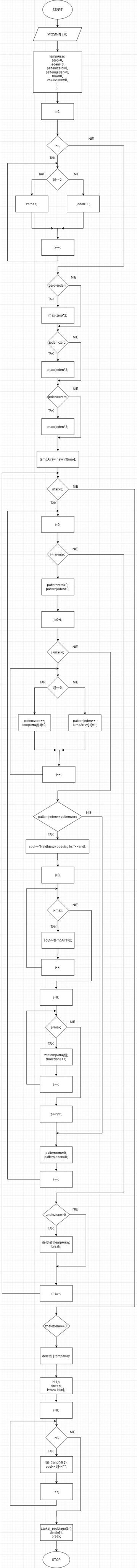
Break

Przypadek 2:test\_1()

Break;

Zwróć 0

### Schemat blokowy



## Opis doświadczenia i dokumentacja

### Zawartość programu

W programie zostały zawarte takie elementy jak:

* Wprowadzenie ciągu wejściowego
* Pętla for obliczająca maksymalną ilość jedynek i zer oraz długość ciągu
* Pętla while służąca do przeszukania ciągu wejściowego (w postaci tablicy)
* Operacja tworząca plik tekstowy, w którym znajdują się wyniki
* Funkcja case pozwalająca na wybór sposobu zbudowania programu

### Problemy i błędy

Podczas pisania programu pojawiły się pewne trudności. Początkowe próby kończyły   
się wieloma niedociągnięciami w kodzie. Zdarzały się braki nawiasów, nadmiar średników oraz niepoprawne zdefiniowanie funkcji. Rozwiązania były niepoprawne lub wybrakowane. Przykładowo: po uruchomieniu, dla tablicy wejściowej wyświetlał się tylko jeden podciąg.   
W niektórych próbach testowych program nie tworzył pliku tekstowego. Po dokładnym przeanalizowaniu problemów i wprowadzeniu znaczących poprawek program działa poprawnie.

# 4.Wnioski

W tym momencie program działa bezbłędnie. Koniecznym było użycie pętli tj. for i while oraz zdefiniowanie funkcji. Aby mieć pewność poprawności należało przeprowadzić kilka testów   
i sprawdzeń. Przy tworzeniu rozwiązania kluczowa jest precyzyjność i dokładna analiza.   
Po wielu nieudanych próbach udało się stworzyć sprawnie działający program. Po uruchomieniu programu wyniki wyświetlają się na konsoli oraz zostają zapisane   
w utworzonym pliku tekstowym.

1. Źródło: https://eduinf.waw.pl/inf/alg/001\_search/0028.php [↑](#footnote-ref-1)