Sprawozdanie

Politechnika rzeszowska im. ignacego łukasiewicza Wydział matematyki i fizyki stosowanej algorytmy i struktury danych

P08

2020

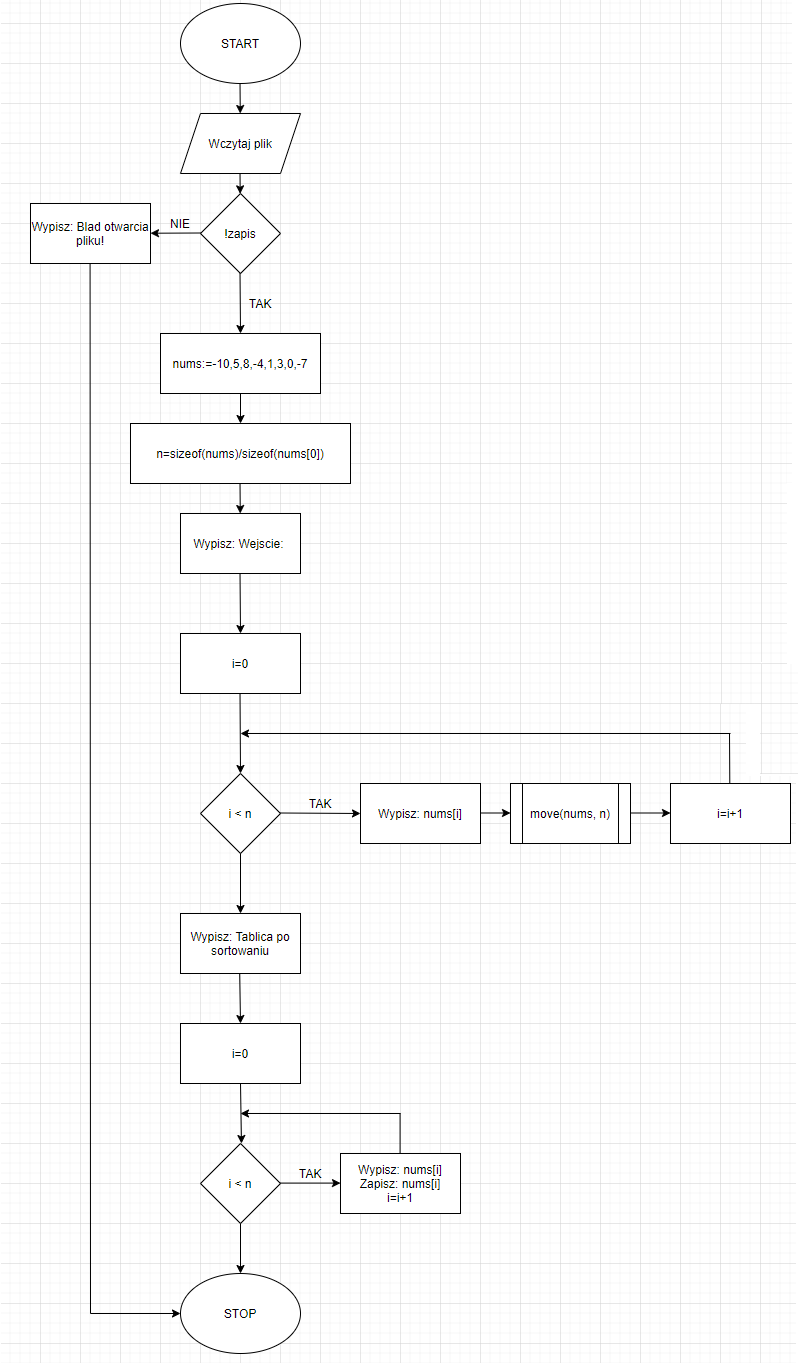
# Wprowadzenie

Moim zadaniem do wykonania było: „dla zadanej tablicy liczb całkowitych przesuń wszystkie elementy mniejsze od 0 na jej koniec (należy zachować kolejność występowania!)”. Do napisania zadania użyłem oczywiście środowiska Code::Blocks IDE i zalecanego języka C++. Zadanie oparte było głównie w pracy na tablicach statycznych w podanym wcześniej języku.

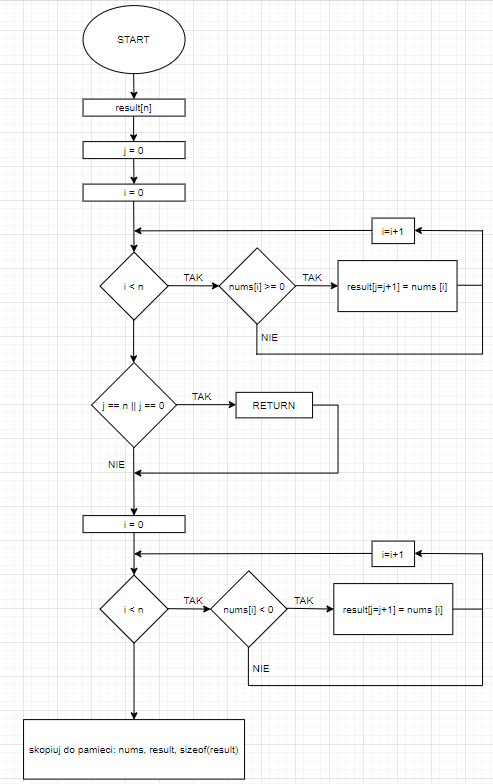
Problem programu polegał na przeniesieniu liczb ujemnych na koniec programu, zachowując przy tym kolejność występowania, czyli liczba (oczywiście ujemna), która była podana jako pierwsza powinna być zaraz za zerem, w związku z tym trzeba było wykorzystać wcześniej wspomniane tablice statyczne i za pomocą napisanej przeze mnie funkcji przerzucić je poza zero oczywiście w kolejności. Podczas pisania programu oprócz wiadomych tablic trzeba było wykazać się znajomością wykorzystania między innymi pętli, w szczególności „for”, gdyż podczas „ustawiania” po kolei liczb trzeba było zmieniać miejsce liczby do pasującego według kolejności, lecz tylko dla danych liczb. Drugim istotnym elementem były na pewno instrukcje warunkowe popularny „if” w moim programie nakazywał co ma zajść jeśli liczba jest mniejsza, większa lub równa oraz oczywiście co ma zostać wykonane jeśli takie warunki zajdą. Wielką potrzebą była umiejętność napisania funkcji oraz zaimplementowania jej w odpowiednim miejscu w kodzie. Osobiście taką funkcje z angielskiego nazwaną „move” zaimplementowałem w kodzie wewnątrz pierwszej pętli for, żeby odpowiednio sortowała zadane wcześniej liczby.

Dzięki poznanej wiedzy wiedziałem jak wykorzystać oddzielnie napisaną funkcję, a później zaimplementować ją w daną linijkę programu. Program poprzez określenie „wielkości” tabeli   
oraz liczby sam potrafi wybrać, w którym miejscu ma występować dana liczba po wykorzystaniu funkcji „move”, następnie kopiując wszystko do pamięci zwraca poprawnie posortowane już liczby w sposób właściwy, jak zadane w poleceniu.

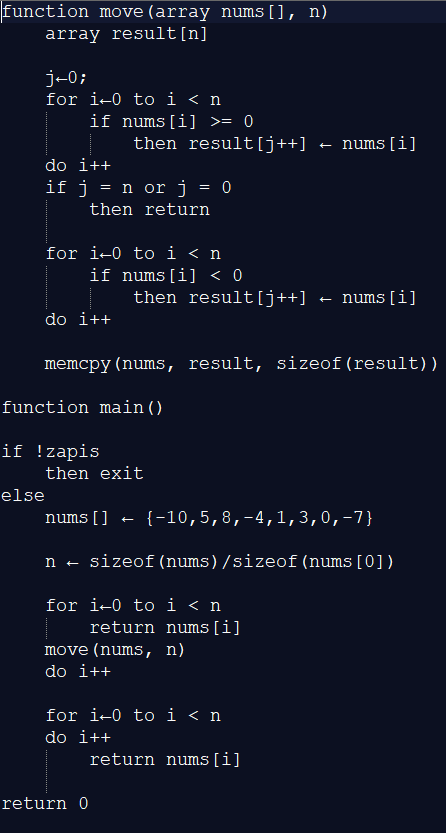
W przelaniu problemu z kartki na program pomogły mi poradniki między innymi Mirosława Zelenta z kanału Pasja Informatyki na serwisie YouTube. W pierwszej kolejności wybrałem te poradniki, żeby nauczyć się podstaw programowania, implementacji tablic oraz pisania podstawowych programów, funkcji i algorytmów. Na drugim miejscu na pewno umieściłbym stronę „stackoverflow”, tam z kolei nauczyłem się wykorzystania wcześniej nie poznanych funkcji w poradnikach wcześniej wymienionego twórcy. Nauczyłem się tam również wyszukiwania potrzebnych informacji   
oraz implementowania we własny kod porad bardziej doświadczonych osób. Dzięki między innymi temu mogłem od podstaw poznać techniki, przydatne funkcje oraz podstawowe jak i bardziej zaawansowane instrukcje jak poprawić swój kod.



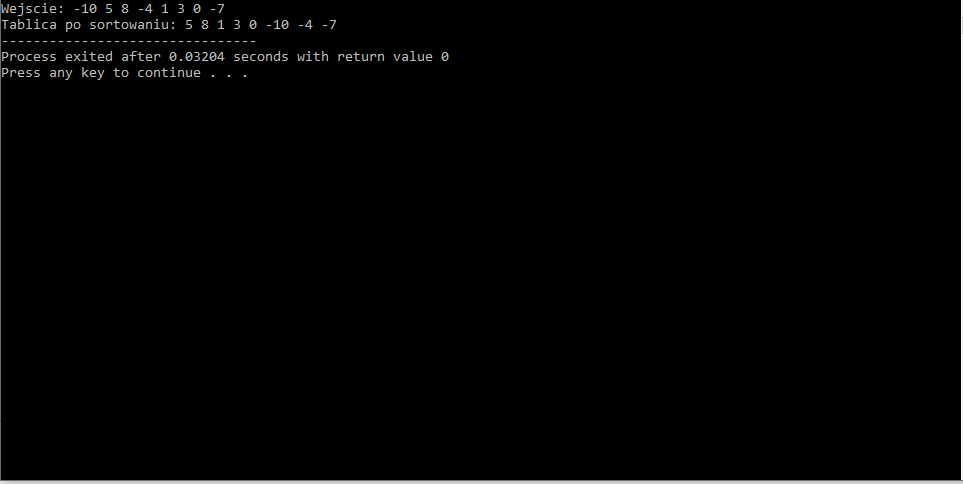
1. *Schemat blokowy funkcji „main”.*



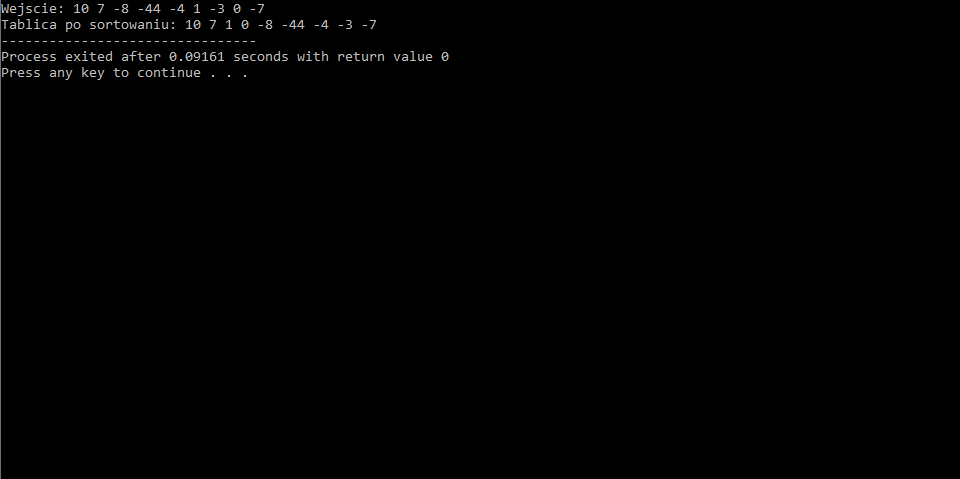
1. *Schemat blokowy funkcji „move”.*



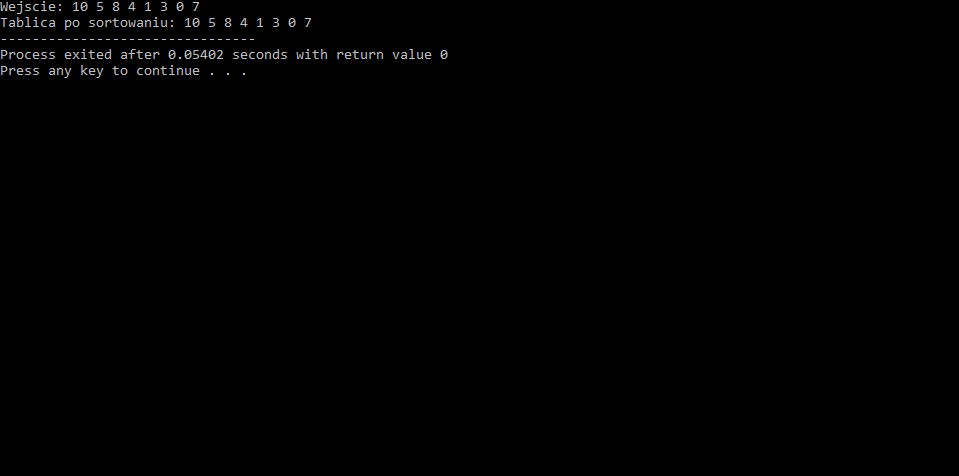
1. *Algorytm w pseudokodzie*.

**

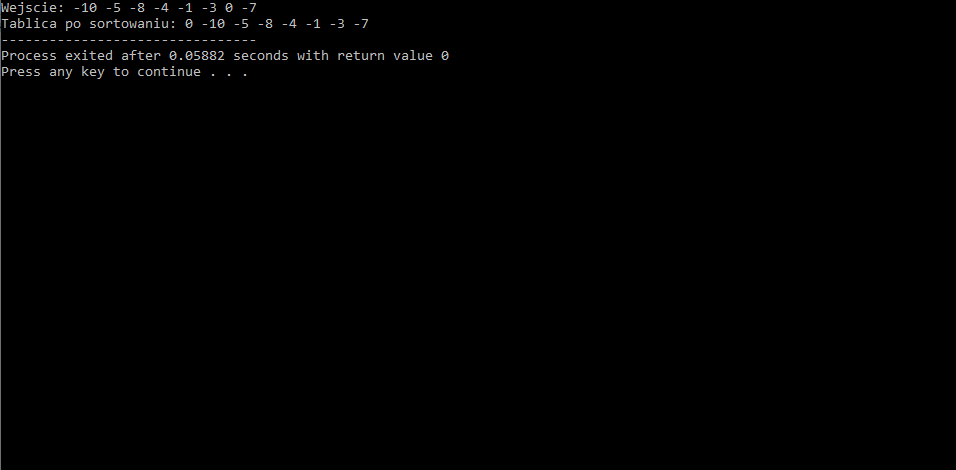
1. *Test z liczbami podanymi w pdf-ie.*

**

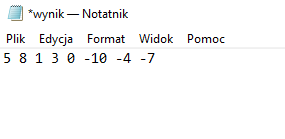
1. *Test z losowymi liczbami.*



1. *Test z samymi liczbami dodatnimi*



1. *Test z samymi liczbami ujemnymi.*



1. *Plik z wynikiem.*

# Wnioski

Struktura programu nie pozwala na jakikolwiek błąd, program wykrywa również wykrywa brak pliku, a algorytm zawsze odpowiednio posortuje każdą zadaną mu tablicę. Zaimplementowana funkcja jak i kod nie wykazywały żadnych problemów w żadnym z testowanych kompilatorów, sortując przy tym tablice odpowiednio.

# Podsumowanie

Program sam w sobie napisany dość prosto, dzięki czemu może służyć jako przykład do nauki w dalszej części edukacji, a jego funkcja segregująca może zostać użyta w innych programach wykorzystujących tablice. Dzięki zadaniu nauczyłem się wyszukiwać błędy w kodzie, wyszukiwać informacje na forach internetowych, oraz poznałem podstawowe jak i bardziej zaawansowane elementy programowania. Zadanie otworzyło mi oczy na możliwości programowania i implementacji pomysłów swoich, jak i innych na kod.