

|  |  |
|--|--|
| <p>Politechnika Świętokrzyska w Kielcach<br/>Wydział Elektrotechniki, Automatyki i Informatyki</p> |  |
| <p>Podstawy Programowania 2 - Projekt<br/>Informatyka- I rok, Rok akademicki - 2021/2022</p>       |  |
| <p>Temat projektu:<br/><b>Sortowanie przez proste wstawianie i metodą Shella</b></p>               | <p>Wykonali:<br/>Maciej Cedzyński<br/>Tomasz Bogdan<br/>Filip Borowiec<br/>Grupa: 1ID11A</p> |

Celem naszego projektu było przygotowanie programu pozwalającego na analizę i obserwację dwóch algorytmów sortowania, Sortowanie przez wstawianie i Sortowania metodą shella.

Na samym początku zapoznaliśmy się z teorią i działaniem tych dwóch algorytmów

- **Sortowanie przez wstawianie** jest jednym z prostszych algorytmów sortowania. Polega on na wybieraniu kolejnych elementów tabeli i przesuwania ich bliżej początku, aż do napotkania elementu o mniejszej wartości. Posiada on mało optymalną złożoność czasową  $O(n^2)$ , natomiast jest on wydajny w przypadku danych już wstępnie posortowanych.
- **Sortowania Shella** jest sortowaniem, które jest po części ulepszoną wersją sortowania przez wstawianie z tą różnicą, że ten algorytm zaczyna od sortowania elementów ustawionych w większych odstępach, a następnie stopniowo zmniejsza odstęp i sortuje ponownie dla tych odstępów. W tym przypadku Shell sort dokonuje sortowania na odstępach równych połowie ilości elementów tabeli, a następnie zmniejsza się o połowę, aż osiągnie sortowanie klasyczne przez wstawianie o 1 element, z tą różnicą, że ze względu na to, że elementy, a już wstępnie posortowane sortowanie nie będzie trwało tak długo.

Po zapoznaniu się z ich działaniem opracowaliśmy implementacje tych algorytmów w programie, do każdego algorytmu opracowaliśmy dwie funkcje jedna odpowiadająca za zwykłe sortowanie i druga która zatrzymuje się na po każdym znaczącym etapie i oczekując wciśnięcie przycisku enter, w ten sposób użytkownik może zaobserwować działanie algorytmu krok po kroku.

Sortowanie w programie jest przeprowadzone na tablicy wypełnionej liczbami typu int. Wypełnienie tablicy liczbami może być przeprowadzone na 3 sposoby.

- odwrócona - tablica jest wypełniona wartościami od 1 do pojemności tablicy ale nie ułożonymi w sposób odwrotny do kierunku sortowania
- pomieszana - tablica jest wypełniona wartościami od 1 do pojemności tablicy ale ich ułożenie w tablicy jest w losowej kolejności

- losowa - tablica jest wypełniona losowymi wartościami od 0 do wartości określonej przez użytkownika

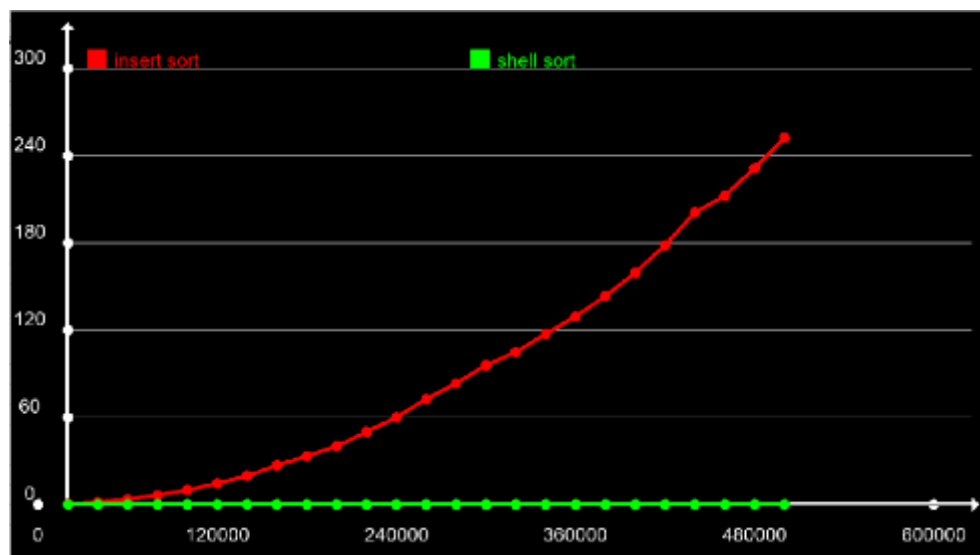
Program oprócz obserwacji pozwala także użytkownikowi zebrać dane o czasie potrzebnym na sortowanie danej ilości danych, działa to na zasadzie określenie co ile danych ma być zbierany czas sortowania i tym samym także wartość początkową a także maksymalną ilość elementów dla jakich ma zbierać dane. Tym samym dla różnych 5000 i maksymalnej wartości 50000 czas sortowania jest generowany dla ilości:

5000, 10000, 15000, 20000, 25000, 30000, 35000, 40000, 45000, 50000

Po wykonaniu wszystkich celów wykorzystaliśmy bibliotekę allegro 5 w celu opracowania przystępnego interfejsu graficznego, w celu kontroli interfejsu opracowaliśmy 2 funkcje. Jedna jest to funkcja która proceduralnej generuje przyciski w stylu okrągłym w zależności od podanych parametrów i druga dająca możliwość wpisywania na ekran liczb z nagłówkiem zależnym od parametru. Dzięki takiemu rozwiązaniu kod jest krótszy i przejrzysty.

W przypadku obserwacji sortowania opracowaliśmy funkcję generującą na ekranie kolumny w zależności od ich wartości, dzięki aktywowaniu tej funkcji po każdym etapie sortowania jesteśmy w stanie w przystępny sposób zaobserwować działanie algorytmów. Podobnie jak w prototypie dodaliśmy możliwość zatrzymywania sortowania po każdym znaczącym etapie w celu obserwacji krok po kroku.

Przy generowaniu danych o czasie sortowania opracowaliśmy wyświetlanie się tych danych w formie przewijanej tabeli wypełniona danymi, następnie program na podstawie tych danych jest w stanie także wygenerować wykres porównujący czas sortowania shella i poprzez wstawianie.

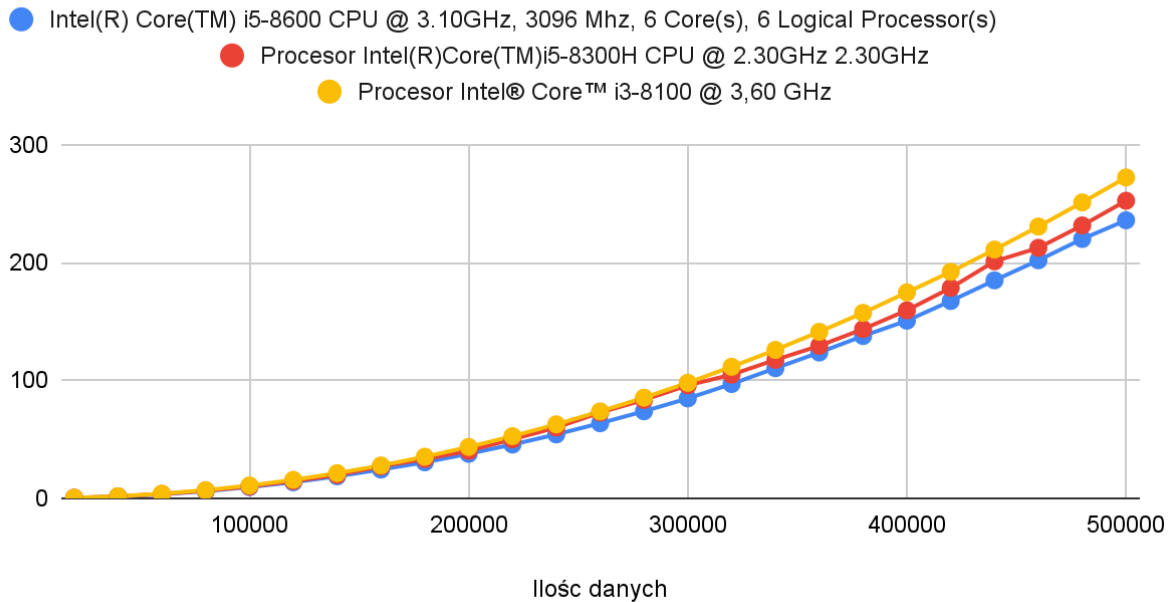


(wykres wygenerowany w programie)

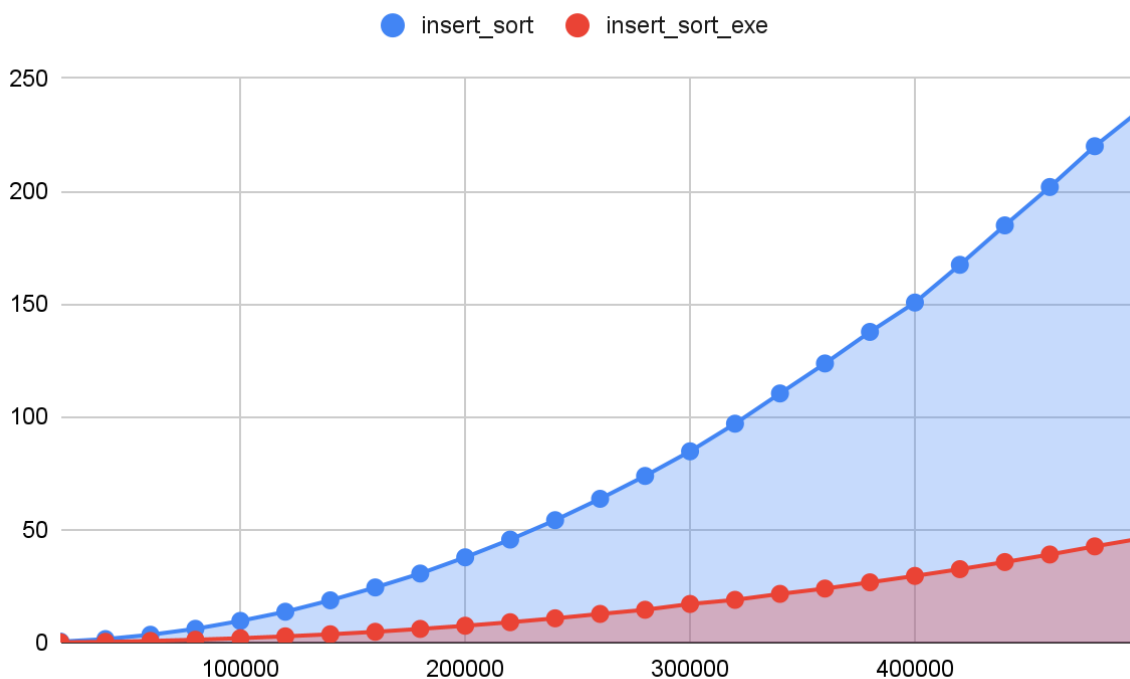
Dane odnośnie czasu sortowania są także zapisywane w pliku tekstowym umożliwiającym przeniesienie danych do arkusza kalkulacyjnego i dokonania głębszej analizy.

W celu przetestowania jak zmienia się prędkość algorytmu w zależności od użytego komputera użyliśmy naszych komputerów do porównania prędkości dla tych samych danych.

## Czas sortowania poprzez wstawianie



Podczas testowania czasu sortowania byliśmy także w stanie zaobserwować ciekawe zjawisko, mianowicie prędkość działania programu otwieranego bezpośrednio w środowisku programistycznym visual studio jest znacznie wolniejsza niż prędkość programu stworzonego z wygenerowanego pliku .exe.



**Link do projektu na GitHub:**

[https://github.com/Filomilo/AiSD\\_Project\\_Insert-SHELL\\_sort](https://github.com/Filomilo/AiSD_Project_Insert-SHELL_sort)  
(branch: master)