***Temat/y: Sortowanie; przez wybór, bąbelkowe, scalanie, wyszukiwanie lidera zbioru oraz wyszukiwanie lidera zbioru poprzez sortowanie.***

***Autor: Piotr Świder***

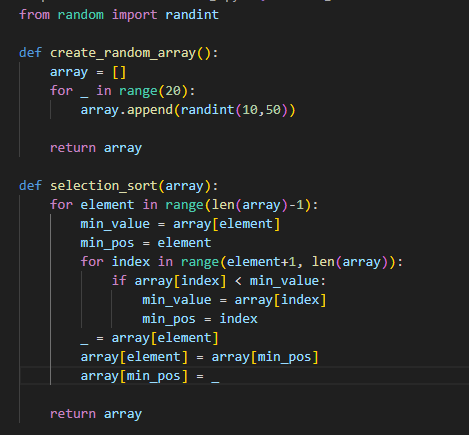
***Grupa laboratoryjna: 8***

***Data: 04/12/2023***

**Zadanie #1.**

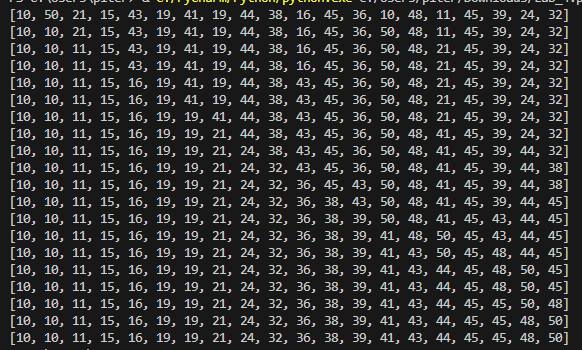
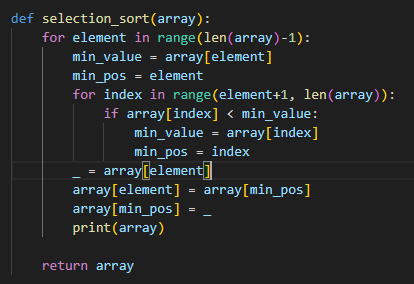
1. Przeanalizować powyższy program, sprawdzić jego działanie;
2. Zmienić liczbę i zakres wylosowanych liczb (N= 20, zakres np. od 10 do 50);
3. Wyświetlić zawartość tablicy po każdym kroku pętli zewnętrznej sortowania;
4. Wyznaczyć wartość minimalną i maksymalną w tablicy;
5. Zmodyfikować program tak, aby posortować tablicę malejąco.

**Rozwiązanie zadania #1.**

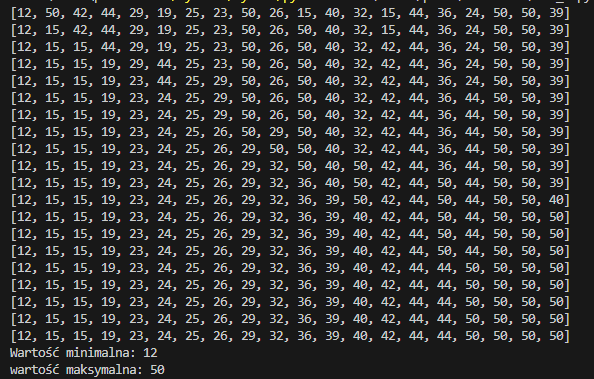
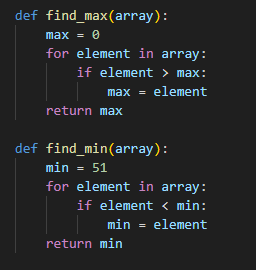
1. Program generuje tablicę o rozmiarze równym N, w której umieszcza pseudolosowe liczby z zakresu od 0 do 199. Wykorzystuję on metodę sortowania przez wybór, która znajduję wartość minimalną, a następnie umieszcza ją na początek tablicy.
2. Kod oraz wartość zwrotna do podpunktu b; 

Komentarz do podpunktu b; Tworzymy funkcję, która tworzy naszą 20-elementową tablicę i uzupełnia ją wartościami pomiędzy 10 a 50.

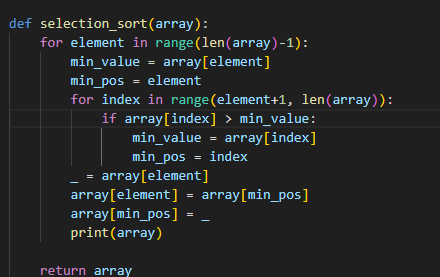
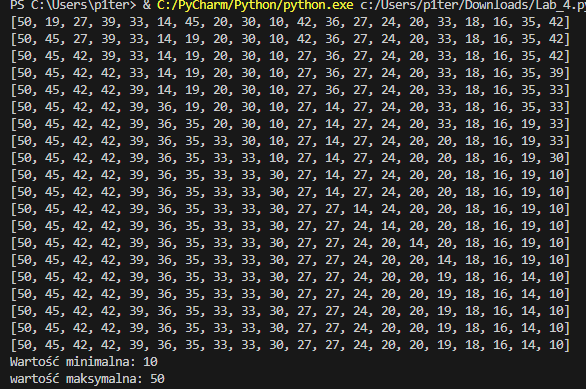
1. Komentarz, kod zwrotny oraz kod programu do podpunktu c; Wystarczy po każdym przejściu pętli zewnętrznej dodać print(array), który będzie wyświetlał każdą pojedynczą zmianę tablicy.



1. Komentarz, kod zwrotny oraz kod programu do podpunktu d; Dodajemy funkcję find\_min oraz find\_max, które szukają wartości minimalnej oraz maksymalnej w zbiorze.



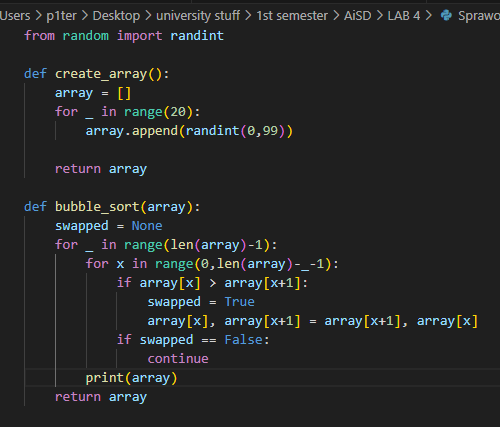
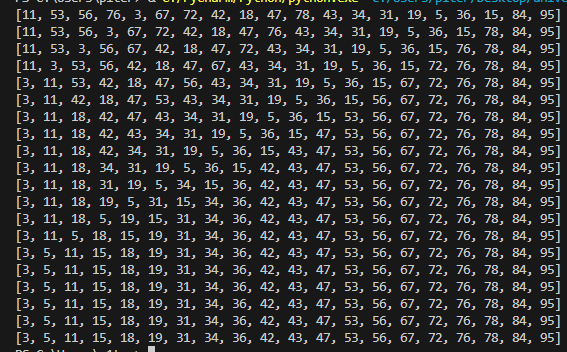
1. Komentarz, kod zwrotny oraz kod programu do podpunktu e; Wystarczy zmienić znak mniejszości w warunku funkcji selection\_sort na znak większości:

**Zadanie #2.**

1. Przeanalizować poniższy program, sprawdzić jego działanie;
2. Wyświetlić zawartość tablicy w każdym kroku pętli zewnętrznej;
3. Iterować pętlę wewnętrzną (z indeksem i ) od N-1 do 0, zaobserwować czy nastąpiła zmiana działania algorytmu. Czy kierunek iteracji ma wpływ na szybkość działania algorytmu?
4. Posortować dane w tablicy malejąco.

**Rozwiązanie zadania #2.**

1. Program przechodzi od końca tablicy i porównuje te – z jej początku oraz zamienia miejscami gdy wartość na większym indeksie jest mniejsza niż wartość na mniejszym indeksie.
2. Kod oraz wartość zwrotna programu do podpunktu b;  
3. Kod oraz wartość zwrotna programu do podpunktu c;
4. Kod oraz wartość zwrotna programu do podpunktu d; 