**Python – jednoczesne znajdowanie minimum i maksimum**

**Przykład 1.**

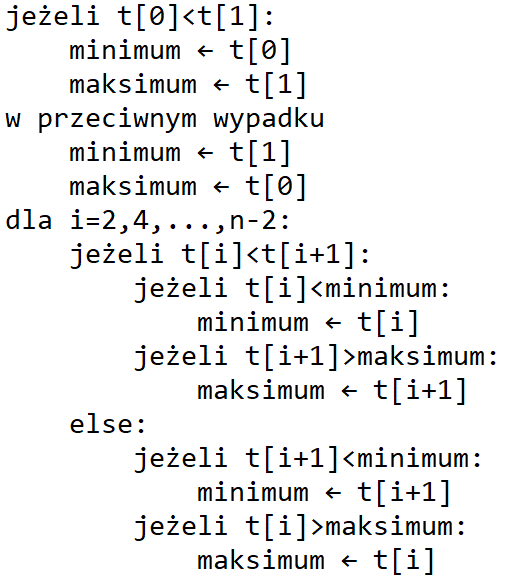
Napisz algorytm w postaci pseudokodu, który znajdzie jednocześnie minimum i maksimum w ciągu liczbowym o parzystej liczbie elementów optymalizując liczbę porównań.

**Specyfikacja:**

**Dane:**t[0..n-1] ciąg liczb całkowitych, gdzie n jest liczbą parzystą

**Wyniki:**minimum – minimum z elementów ciągu t  
maksimum – maksimum z elementów ciągu t

**Pseudokod:**



Komentarz:

W powyższym algorytmie dzielimy cały ciąg na liczby, które są mniejsze w parze sąsiednich elementów do porównywania z dotychczasowym minimum oraz pozostałe liczby do porównywania z dotychczasowym maksimum. Dzięki temu nie musimy osobno porównywać każdej liczby z dotychczasowym minimum oraz maksimum, co optymalizuje liczbę porównań. Na przykład, jeżeli trzecia liczba jest mniejsza niż czwarta, to trzecią sprawdzamy, czy jest mniejsza niż dotychczasowe minimum, a czwartą sprawdzamy, czy jest większa od dotychczasowego maksimum.

**Uwaga:**

W poniższych zadaniach nie korzystaj z wbudowanych funkcji min i max.

**Zadanie 1**

Napisz program w języku Python, który wylosuje 100 liczb całkowitych z przedziału od 1 do 1000 i wyświetli minimum i maksimum spośród wylosowanych liczb. Nie korzystaj z optymalizacji. Podaj ile zostanie wykonanych porównań.

**Zadanie 2**

Napisz ile wykonywanych jest porównań w przykładzie 1.

**Zadanie 3**

Napisz program, który wylosuje 100 liczb całkowitych z zakresu od 1 do 1000 oraz wyświetli minimum i maksimum spośród wylosowanych liczb stosując zoptymalizowany algorytm z przykładu 1. Napisz, ile porównań wykonuje ten program.

**Zadanie 4**

Zmodyfikuj przykład 1, tak aby działał dla ciągu liczb o nieparzystej liczbie elementów.

Wskazówka:

Przypisz początkowo do minimum i maksimum pierwszy wyraz ciągu, a pozostałe porównuj parami.