

Algorytmy i Struktury Danych - 2. rok Inżynierii i Analizy Danych

Laboratoria 2. Sortowanie przez wstawianie, algorytm Hornera

Kolejną prostą metodą sortowania danych jest algorytm sortowania przez wstawianie. W i -tym kroku iteracji pierwszych i elementów sortowanej tablicy jest posortowanych. Bierzemy element $i + 1$ i przesuwamy elementy leżące na lewo od niego o 1 miejsce w prawo, aż natrafimy na element mniejszy od tego $i + 1$. Wówczas wstawiamy go na prawo od tego mniejszego elementu i przechodzimy do kolejnego kroku iteracji.

Sortowanie Przez Wstawianie (Insertion Sort)

Poniżej w nieformalnym języku przedstawiony został algorytm sortowania przez wstawianie zastosowany do tablicy z danymi, porządkowanej w kolejności rosnącej. W algorytmie tym przyjmujemy, że n jest liczbą zawartych w tablicy sortowanych danych, a A tablicą zawierającą te dane (zakładamy, że jest ona numerowana od 1 do n).

```
Powtarzaj dla i od 1 do (n-1)
{
  Wybierz element A[i+1] i wstaw go do zmiennej a
  wstaw i do zmiennej j
  Dopoki ((j>0) i (A[j]>a))
  {
    Wstaw A[j] do A[j+1]
    zmniejsz j o 1
  }
  Wstaw a do A[j+1]
}
```

Algorytm Hornera

Algorytm Hornera służy do zoptymalizowanego wyznaczania wartości wielomianu w danym punkcie x_0 . Wykonując takie obliczenie po kolei dla wielomianu stopnia n , o współczynnikach a_n przy n -tych potęgach:

$$W(x_0) = a_n * \underbrace{x_0 * \dots * x_0}_{n \text{ razy}} + a_{n-1} * \underbrace{x_0 * \dots * x_0}_{(n-1) \text{ razy}} + \dots + a_1 * x_0 + a_0,$$

wykonujemy n dodawań i $n + (n - 1) + \dots + 1 = \frac{n(n+1)}{2}$ mnożeń. Jeśli jednak wykonamy to obliczenie zgodnie z poniższym schematem (schematem Hornera):

$$W(x_0) = (\dots (((a_n * x_0 + a_{n-1}) * x_0 + a_{n-2}) * x_0 + a_{n-3}) * x_0 + \dots) * x_0 + a_0,$$

wówczas musimy wykonać jedynie n dodawań i n mnożeń

Zadanie 1.

Zdefiniuj klasę o nazwie `Wiel` pozwalającą przechowywać wielomiany, jak również wykonywać na nich podstawowe operacje: tworzenie wielomianu, usuwanie wielomianu, ustalanie i wydawanie współczynników i wartości wielomianu dla podanego argumentu, drukowanie wielomianu; w szczególności klasa ta powinna zawierać też metodę:

```
bool Wiekszy(Wiel *wielomian2);
```

służącą do sprawdzenia czy wartość w ostatnio wyznaczonym punkcie wielomianu, reprezentowanego przez obiekt wywołujący tę metodę jest większa od wartości wielomianu reprezentowanego przez obiekt na który wskazuje wskaźnik będący argumentem tej metody (wielomian2). W celu unikania wielokrotnego wyznaczania wartości wielomianu dodaj pole przechowujące wartość wielomianu w ostatnio wybranym punkcie.

Napisz program pobierający od użytkownika następujące dane:

- 1) Liczba całkowita (n), której wartość będzie równa podanej przez użytkownika liczbie wielomianów.
- 2) W n -krotnej pętli program pobiera od użytkownika stopień wielomianu, oraz w kolejnej pętli liczby rzeczywiste będące kolejnymi współczynnikami danego wielomianu.
- 3) Wartość rzeczywistą punktu, w którym porównywane będą wartości wielomianów.

Program zwraca podane wielomiany posortowane, metodą sortowania przez wstawianie, rosnąco co do ich wartości w wybranym punkcie, oraz wartości tych wielomianów w tym punkcie.

Uwaga:

W programie należy zastosować dynamiczne tworzenie obiektów klasy `Wiel` jak również dynamicznie utworzyć tablice wskaźników do tych obiektów, oraz tablicę współczynników wielomianu.