

## בניית שרשרת חוליות והדפסתה

(שרשרת חוליות = רשימה מקושרת דינאמית)

### I בנייה מהסוף להתחלה

בסיום הפעולה הערך הראשון יהיה בסוף השרשרת והאחרון בתחילתה.

בניית שרשרת (באורך n)

- .1  $lst \leftarrow \text{null}$
- .2 בצע n פעמים:
  - 2.1 קלט x
  - 2.2 יצירת חוליה עם x וערך lst
- .3 החזר את lst

```
//--- בניית שרשרת חוליות מהסוף להתחלה ---  
//--- האיבר האחרון יהיה בתחילת השרשרת ---  
public static Node<int> Build(int n)  
{  
    Node<int> lst = null;  
    int x;  
  
    for (int i = 0; i < n; i++)  
    {  
        Console.WriteLine("type a value --> ");  
        x = int.Parse(Console.ReadLine());  
  
        lst = new Node<int>(x, lst);  
    }  
    return lst;  
}
```

## II בנייה מהתחלה לסוף

בסיום הפעולה הערך הראשון יהיה בתחילת השרשרת והאחרון בסופה.

## בניית שרשרת (באורך n)

- .1  $lst \leftarrow null$
- .2  $pos \leftarrow null$
- .3 בצע n פעמים:
  - 3.1 קלט x וצור חוליה עם x  $temp \leftarrow x$
  - 3.2 אם  $(lst == null)$  // זוהי חוליה ראשונה
    - 3.2.1  $lst \leftarrow null$  ועם x ועם null
    - 3.2.2  $pos \leftarrow lst$
  - 3.3 אחרת // זוהי חוליה שאינה ראשונה
    - 3.3.1 עדכן את next של pos לקבל חוליה חדשה עם x ועם null
    - 3.3.3 קדם את pos לחוליה החדשה
- .4 החזר את lst

```

//--- בניית שרשרת חוליות מהתחלה לסוף ---
//--- המספר הראשון שנקלט יהיה ראשון בשרשרת, והאחרון יהיה אחרון בשרשרת ---
public static Node<int> Build(int n)
{
    int x;
    Node<int> lst = null;
    Node<int> pos = null;

    for (int i = 0 ; i < n ; i++)
    {
        Console.WriteLine("type 1st value --> ");
        x = int.Parse(Console.ReadLine());

        if (lst == null)
        {
            lst = new Node<int>(x);
            pos = lst;
        }
        else
        {
            pos.SetNext(new Node<int>(x));
            pos = pos.GetNext();
        }
    }
    return lst;
}

```

## III הדפסת איברי השרשרת

כשסורקים שרשרת מותר לשנות את ערך ההפנייה לתחילת השרשרת, כי קיימת הפנייה לתחילת השרשרת מהפעולה המזמנת.

```
public static int showChain (Node<Integer> lst)
{
    :
    while (lst != null)
    {
        : // lst.getValue() בצע משהו עם/על
        lst = lst.getNext();
    }
    return ...
}
```

```
///--- פעולה המקבלת שרשרת של מספרים שלמים ומציגה אותה ---
public static void Show(Node<int> lst)
{
    Console.WriteLine("[");
    Node<int> pos = lst;
    while (pos != null)
    {
        Console.WriteLine(pos.ToString());
        if (pos.HasNext())
            Console.Write(", ");
        pos = pos.GetNext();
    }
    Console.WriteLine("]");
}
```

פלט הפעולה יהיה השרשרת במבנה:  $[x_1, x_2, \dots, x_n]$

כך ש-  $x_1$  הוא האיבר שבראש השרשרת ו-  $x_n$  בסופה.