

# برمجة متقدمة - ١ -

Collections - ٢ -

# الارتال

## QUEUES

### المحاضرة السادسة

# الرتل

## QUEUE CLASS

- تستخدم عندما تريد حفظ بيانات بحيث يكون "الداخل أولاً خارجاً أولاً"
- عند الحذف من الرتل تستخدم الدالة **deque**
- عند الاضافة الرتل تستخدم الدالة **enqueue**

# Queue

- الرتل بمعنى اخر هو اشبه بطابور انتظار وهذا الطابور يزداد حجمه بإضافة البيانات في آخره ويقل بإخراج البيانات من الأمام.
- المعنى العام شبيه بالطوابير العادية حيث أن من يأتي أولاً يخرج أولاً والمتأخر يقف في آخر الصف
- إذا على عكس stack فإن Queues لديه نهاية ومقدمة
- النهاية لدخول البيانات والمقدمة لخروجها.
- كتشبيه عنه طابعة الورق ...

## أين نستخدم Queue

- نستخدم في تطبيقات خدمة العملاء فعندها نضيف آخر واصل في آخر الرتل ونقوم بخدمة الأول واخراجه من الرتل.
- نستخدم في ترتيب العمليات في CPU بعض الأحيان يستعان بـ Queue تقدم المهم أولاً.
- الاستعلام عن العناصر في Tree وهي عبارة عن طريقة لحفظ البيانات كما القوائم والمصفوفات.

التسلسل	دوال الرتل	الاستخدام
1	void Clear();	ازالة كل العناصر من الرتل
2	bool Contains( object obj );	تحديد إذا ما كان العنصر في الرتل
3	object Dequeue();	ازالة وارجاع العنصر من بداية الرتل
4	void Enqueue( object obj );	اضافة عنصر الى نهاية الرتل
5	void TrimToSize();	كتابة سعة العناصر الفعلية في الرتل
6	object[] ToArray();	نسخ الرتل الى مصفوفة
التسلسل	الخاصية	الاستخدام
1	Count	قراءة عدد العناصر التي يحتويها الرتل

```
using System;
using System.Collections;
namespace CollectionsApplication
{
    class Program
    {
        static void Main(string[] args)
        {
            Queue q = new Queue();
            q.Enqueue('A');
            q.Enqueue('M');
            q.Enqueue('G');
            q.Enqueue('W');
```

```
Console.WriteLine("Current queue: ");
    foreach (char c in q)
        Console.Write(c + " ");
    Console.WriteLine();
    q.Enqueue('V');
    q.Enqueue('H');
    Console.WriteLine("Current queue: ");
    foreach (char c in q)
        Console.Write(c + " ");
    Console.WriteLine();
    Console.WriteLine("Removing some values ");
    char ch = (char)q.Dequeue();
    Console.WriteLine("The removed value: {0}", ch);
    ch = (char)q.Dequeue();
    Console.WriteLine("The removed value: {0}", ch);
    Console.ReadKey();
} }
```



Current queue:

A M G W

Current queue:

A M G W V H

Removing values

The removed value: A

The removed value: M

# الخلاصة:

- المكدرات والارتال من أنواع هياكل البيانات **Data Strucutre** يستخدمان لتخزين المعلومات في الذاكرة وكل نوع له اليه في التعامل مع البيانات
  - Stack يستخدم LIFO Last In First Out
  - Queue يستخدم FIFO First In First Out

????????? •