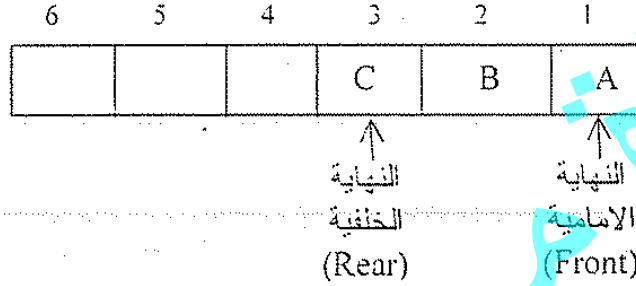
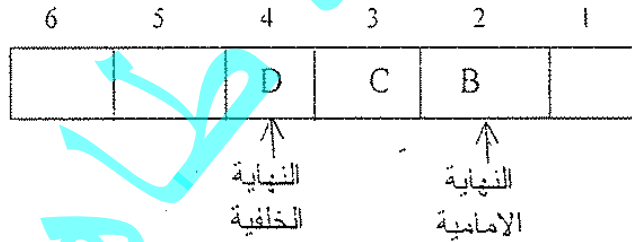


### 3-3 الطابور Queue

هو هيكل تسلسلي (sequential) تكون فيه عمليات الاضافة في النهاية الخلفية (Rear) وعمليات الحذف في النهاية الاخرى (الامامية Front) كما في الشكل التالي :



حيث نلاحظ ان العنصر (A) في مقدمة الطابور يليه العنصر (B) ثم (C) وعند اضافة عنصر جديد يكون موقعه بعد (C) ، اما عند حذف عنصر من الطابور تكون عملية الحذف من النهاية الامامية أي حذف العنصر (A) ويصبح الشكل اعلاه بعد اضافة العنصر (D) وحذف العنصر (A) كالآتي :-



نجد ان الطابور يفيد في الفعاليات التي تتضمن جدولة الاعمال حسب ترتيب وصولها او طلبها ويمكن تلخيص هذا بالعبارة الاتية :

اول من يدخل اول من يخرج [ First IN First Out ( FIFO ) ]  
أي ان من يصل اولاً يحصل على الخدمة اولاً وتسمى عملية الاضافة الى الطابور (ENQueue) او (Insertion) اما عملية الحذف فتسمى (DEQueue) او (Deletion) .

### 3-3-1 تمثيل الطابور باستخدام المصفوفة

#### Array Representation of Queue

يطبق الطابور باستخدام مصفوفة احادية بالسعة المطلوبة (SIZE) وبالنوع المناسب لنوع البيانات (Data Type) التي ستخزن فيه ( Real , Int. ... الخ ) مع استخدام :

المتغير (Rear) كمؤشر يشير الى موقع العنصر الاخير في الطابور  
 المتغير (Front) كمؤشر يشير الى موقع العنصر الاول في الطابور  
 ان قيمة المؤشرين في الحالة الابتدائية (  $Rear = 0$  ,  $Front = 0$  ) عندما يكون الطابور خاليا (Empty) من العناصر  
 تنفذ عملية اضافة عنصر الى الطابور بعد تحديث قيمة المؤشر (Rear) ليشير الى الموقع الجديد بعد موقع اخر عنصر اما تنفيذ عملية حذف عنصر من الطابور فيحدث المؤشر (Front) ليشير الى موقع العنصر التالي بعد حذف العنصر في المقدمة ولنفترض لدينا الطابور (Q) سعته (6) عناصر وننفذ عليه سلسلة العمليات الآتية :-

#### حالة الطابور

الحالة	المؤشر F	المؤشر R	Q [1]	Q [2]	[3]	[4]	[5]	[6]
- الطابور خالي	0	0	-	-	-	-	-	-
- اضافة العنصر A	1	1	A	-	-	-	-	-
- اضافة العنصر B	1	2	A	B	-	-	-	-
- اضافة العنصر C	1	3	A	B	C	-	-	-
- حذف عنصر	2	3	-	B	C	-	-	-
- اضافة العنصر D	2	4	-	B	C	D	-	-
- اضافة العنصر E	2	5	-	B	C	D	E	-
- حذف عنصر	3	5	-	-	C	D	E	-
- حذف عنصر	4	5	-	-	-	D	E	-