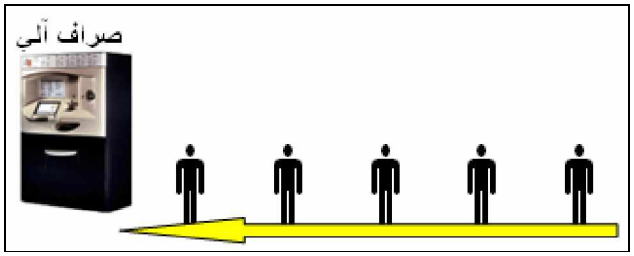
الطوابير (الصفوف)

Queues

1- تعريف

الطابور أو الصف مجموعة من العناصر المتواجدة على خط واحد بحيث تتم خدمة العنصر الأول، ثم الثاني، الخ ... أما العناصر الجديدة فتضاف دائما إلى آخر الصف.

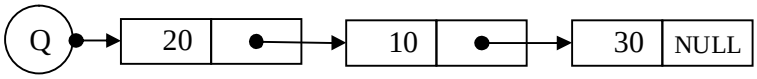
مثال: طابور العملاء أمام الصراف.



أول عنصر يدخل إلى الصف هو أول عنصر يخرج منه مما يعني أن معالجة عناصر الصف تتم حسب مبدأ (First In First Out - FIFO).

2- تمثيل الصف

تعتبر الصفوف نوع خاص من القوائم المتصلة التي تطبق فيها عمليات الإضافة في آخر القائمة بينما تتم عمليات الحذف دائما على العنصر الأول (رأس القائمة).



في لغة C++ يمكن الإعلان عن الصفوف كما يلي:

```
struct Node
{
        int Item;
        Node  *Next;
};

Node  *Q;
```

تتم معالجة عناصر الصف عن طريق الدوال الخمس التالية:

الدالة	دورها
Node* InitQueue(Node *Q)	استهلال الصف Q ليكون فارغا في البداية
bool IsEmpty(Node *Q)	التحقق إذا ما كان الصف Q فارغا أم لا
Node * EnQueue(int x, Node *Q)	إضافة العنصر x إلى آخر الصف Q
Node * DeQueue(Node *Q)	حذف العنصر الأول من الصف Q
void Print(Node *Q)	طباعة عناصر الصف Q على الشاشة

يقوم البرنامج التالي بإنشاء صف يحتوي على 5 عناصر ثم يقوم بحذف العنصر الأول وعرض بقية العناصر على الشاشة.

```
#include <iostream>
using namespace std;

struct Queue
{
     int Item;
     Queue *Next;
};
Queue* InitQueue(Queue *Q)
{
     Q = NULL;
     return Q;
}
bool IsEmpty(Queue *Q)
{
     if (Q == NULL)
       return true;
     else
       return false;
}
Queue* EnQueue(int x, Queue *Q)
{
     if (IsEmpty(Q))
     {
          Queue *R = new(Queue);
          R->Item = x;
          R->Next = NULL;
          Q = R;
     }
     else
     {
          Queue *P = Q;
          while(P->Next != NULL)
               P = P->Next;
          Queue *R = new(Queue);
```

```
        R -> Item = x;
        R->Next = NULL;
        P->Next = R;
    }
    return Q;
}
Queue*  DeQueue(Queue  *Q)
{
    if (!IsEmpty(Q))
    {
        Queue  *P = Q;
        Q = Q->Next;
        delete P;
    }
    return Q;
}
void  Print(Queue *Q)
{
    Queue  *P = Q;
    while (!IsEmpty(P))
    {
        cout<<P->Item<<"\t";
        P = P-> Next;
    }
}
void main()
{
    Queue *Q = NULL;
    Q = InitQueue(Q);
    for (int i = 1; i<= 5; i++)
    {
        Q = EnQueue(i, Q);
    }
    Q = DeQueue(Q);
    Print(Q);
    cout<<endl;
}
```

مخرجات البرنامج

```
2       3       4       5
Press any key to continue_
```

تمرين رقم 1

قم بإكمال البرنامج التالي الذي يقوم بإنشاء تصنيف (class) لمعالجة الصفوف:

```
#include <iostream>
using namespace std;

class Queue
{
        private:
                int Item;
                Queue  *Next;
        ……….
                Queue* InitQueue();
                bool IsEmpty();
                Queue*  EnQueue(int);
                ……..;
                void  Print();
};
Queue* Queue::InitQueue()
{
        ……..;
}
bool Queue::IsEmpty()
{
        if (this==NULL)
                return true;
        else
                return false;
}
Queue*  Queue::EnQueue(int x)
{
        if  (this->IsEmpty())
        {
                Queue  *R = new(Queue);
                R->Item =  x;
                R->Next = NULL;
                return R;
        }
        else
        {
                Queue *T = this;
                Queue  *P = this;
                while(P->Next != NULL)
                        P = P->Next;
                Queue  *R = new(Queue);
                R -> Item = x;
                R->Next = NULL;
                P->Next = R;
                return T;
        }
}
Queue*  Queue::DeQueue()
{
```

```
        if (!this->IsEmpty())
        {
            Queue  *P = this;
            Queue *T = this->Next;
            delete P;
            return T;
        }
        else
            ……;
}
void  Queue::Print()
{
        Queue  *P = this;
        while (!P->IsEmpty())
        {
            cout<<P->Item<<"\t";
            ……;
        }
}
void main()
{
        Queue *Q;
        ……Q->InitQueue();
        for (int i = 1; i<= 5; i++)
        {
            …… Q->EnQueue(i);
        }
        ……Q->DeQueue();
        Q->Print();
        cout<<endl;
}
```

تمرين رقم 2

اكتب دالة Size تقوم بحساب عدد عناصر الصف Q.

تمرين رقم 3

اكتب دالة Invert تقوم بقلب عناصر الصف Q.