

به نام خدا

ساختمان داده ها و الگوریتم ها

محمد مهدی گیلانیان صادقی

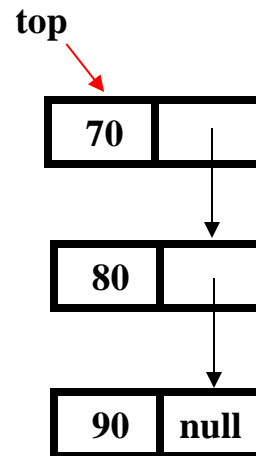
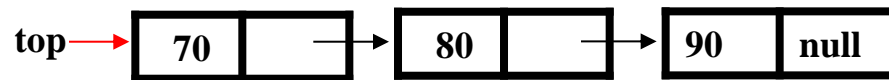
دانشگاه آزاد اسلامی واحد قزوین

نیمسال دوم ۱۴۰۱-۱۴۰۲



• پیاده سازی پشته با لیست پیوندی:

```
class node {  
    friend class stack;  
private:  
    int info;  
    node *next;  
};  
class stack{  
    public:  
        // member functions  
    private:  
        node *top;  
};
```





عمل ایجاد پشته:

```
stack::stack(){  
top = null;  
}
```



عمل ایجاد پشته:

```
stack::stack(){  
top = null;  
}
```

عمل تست خالی بودن پشته:

```
int stack::empty(){  
if (top == null)  
    return 1;  
return 0;  
}
```



صفحه ۴

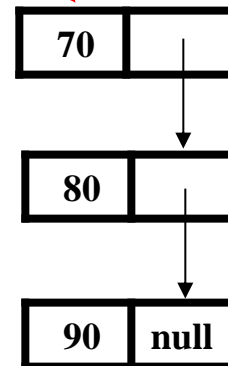
ساختمان داده ها و الگوریتم ها

عمل اضافه کردن به پشته (push):

top
↓
null

الف

top



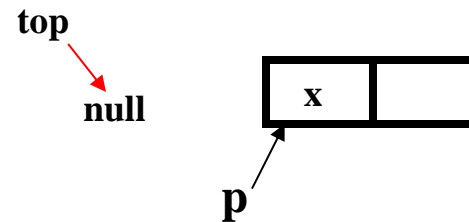
ب



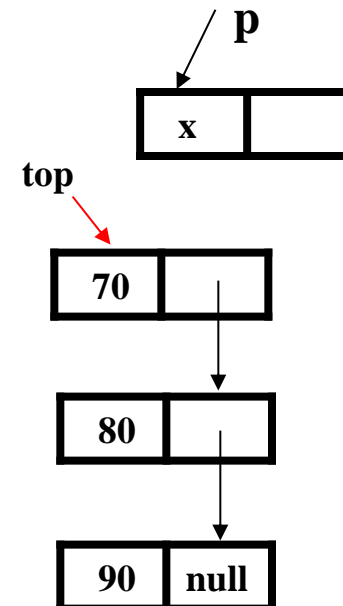
عمل اضافه کردن به پشته (push):

```
void stack::push(int x)
{
    node *p;
    p= new node;
    p->info=x;

}
```



الف

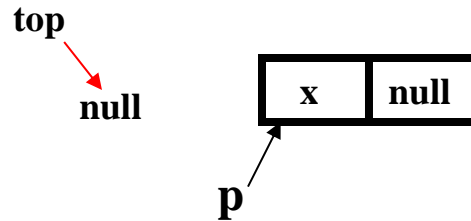


ب

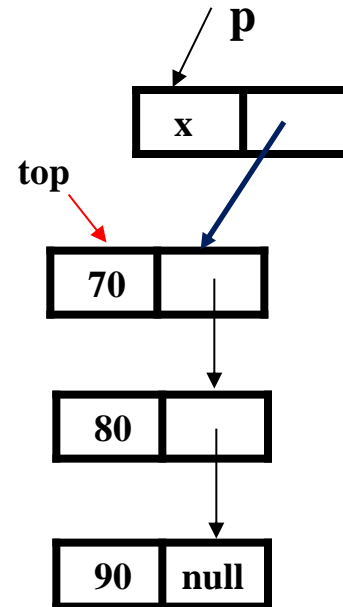


عمل اضافه کردن به پشته (push):

```
void stack::push(int x)
{
    node *p;
    p= new node;
    p->info=x;
    p->next=top;
}
```



الف

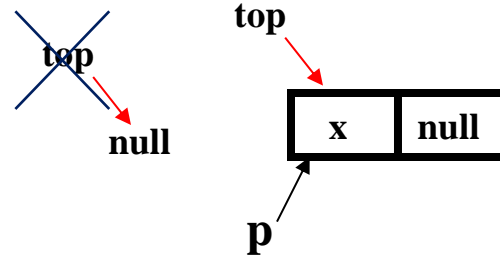


ب

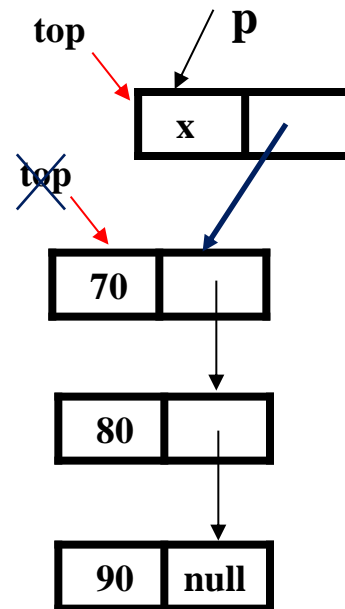


عمل اضافه کردن به پشته (push):

```
void stack::push(int x)
{
    node *p;
    p= new node;
    p->info=x;
    p->next=top;
    top=p;
}
```



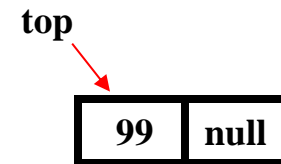
الف



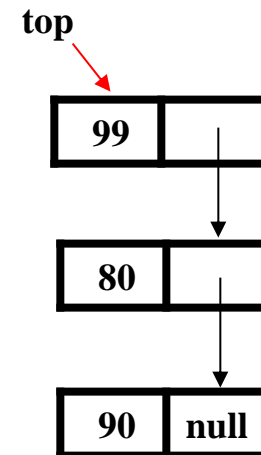
ب



عمل حذف کردن از پشته (pop):



الف



ب

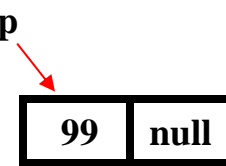


ساختمان داده ها و الگوریتم ها

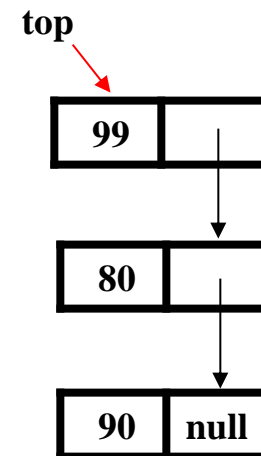
صفحه ۹

عمل حذف کردن از پشته (pop):

```
void stack::pop(int &x, int &underflow){  
    node *p;  
    underflow = 0;  
    if(empty()){  
        underflow = 1;  
        cout<<"stack is empty";  
    }  
}
```



الف



ب

}



عمل حذف کردن از پشته (pop):

```
void stack::pop(int &x, int &underflow){
```

```
node *p;
```

```
underflow = 0;
```

```
if(empty()){
```

```
    underflow = 1;
```

```
    cout<<"stack is empty";
```

```
}
```

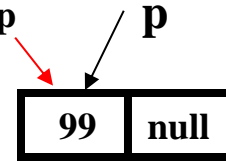
```
else{
```

```
    x=top->info;
```

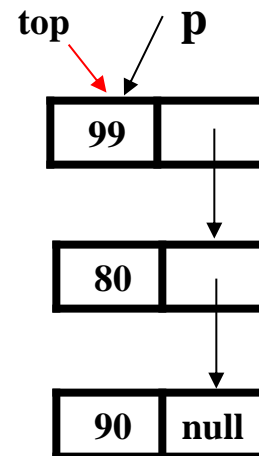
```
    p=top;
```

```
}
```

```
}
```



الف

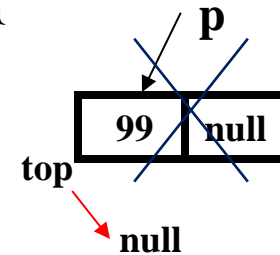


ب

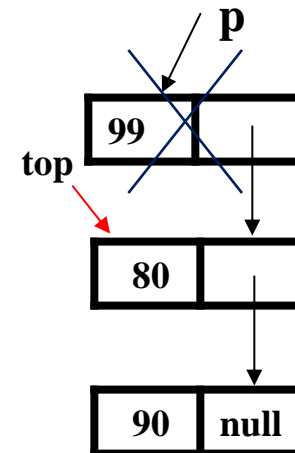


عمل حذف کردن از پشته (pop):

```
void stack::pop(int &x, int &underflow){  
    node *p;  
    underflow = 0;  
    if(empty()){  
        underflow = 1;  
        cout<<"stack is empty";  
    }  
    else{  
        x=top->info;  
        p=top;  
        top=top->next;  
        delete p;  
    }  
}
```



الف

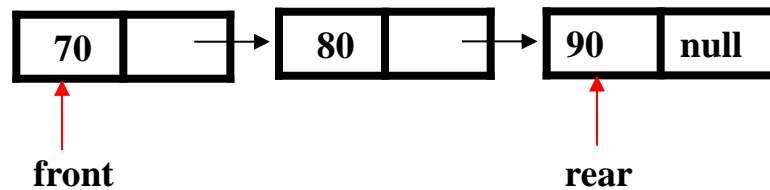


ب



پیاده سازی صف با لیست پیوندی:

```
class node{  
    friend class queue;  
private:  
    int info;  
    node *next;  
};  
class queue{  
    public:  
        // member functions  
    private:  
        node *front;  
        node *rear;  
};
```





عمل ایجاد صف:

front = rear = null;



عمل ایجاد صف:

```
front = rear = null;
```

عمل تست خالی بودن صف:

```
if (front == null)  
    // صف خالی است
```



صفحه ۱۵

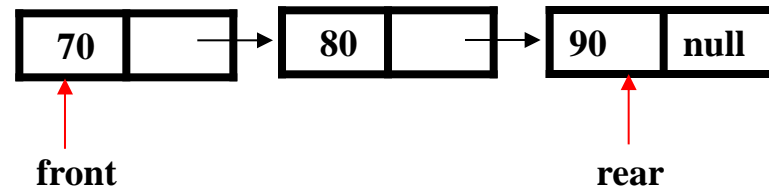
ساختار داده ها و الگوریتم ها

عمل افزودن به صف (addq):

الف

front
↑
null

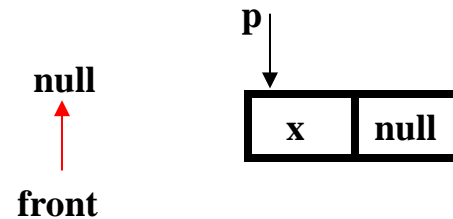
ب





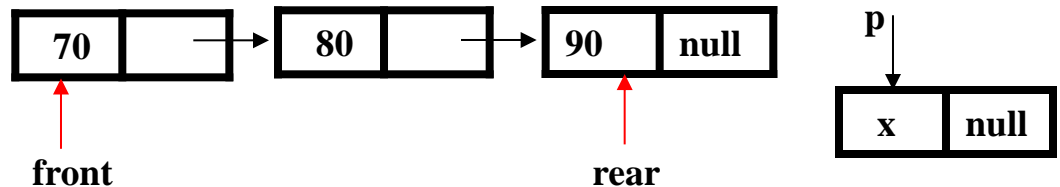
عمل افزودن به صف (addq):

```
node *p;  
p= new node;  
p->info = x;  
p->next = null;
```



الف

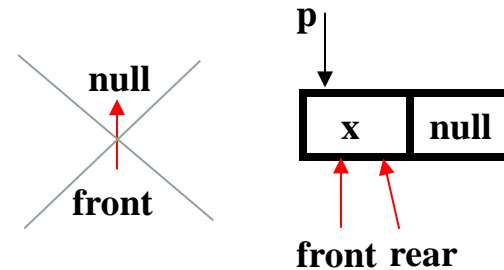
ب





عمل افزودن به صف (addq):

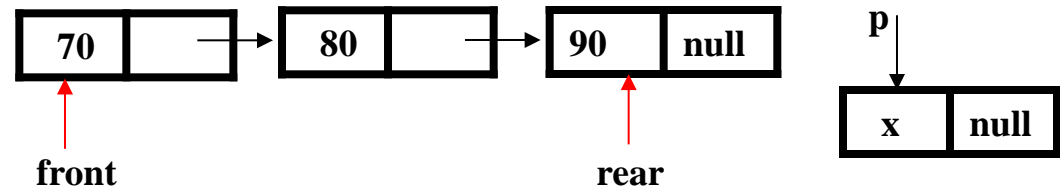
```
node *p;  
p= new node;  
p->info = x;  
p->next = null;  
if (front == null)  
    front = p
```



الف

ب

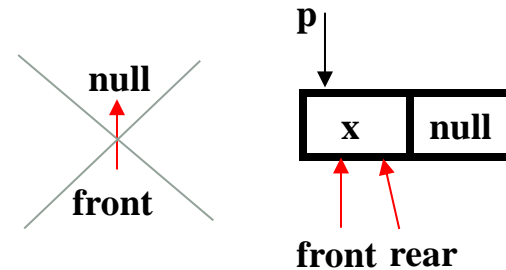
```
rear = p
```





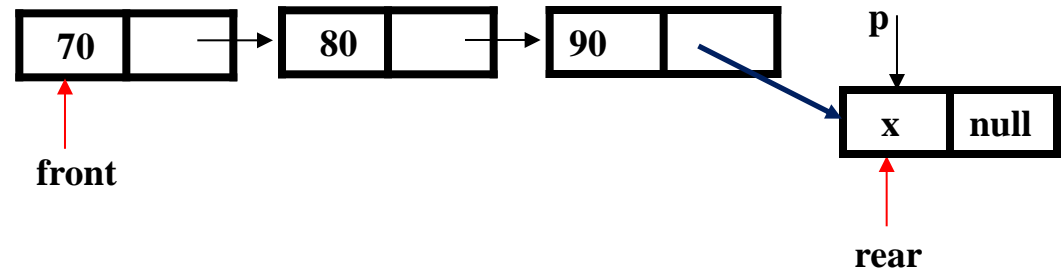
عمل افزودن به صف (addq):

```
node *p;  
p= new node;  
p->info = x;  
p->next = null;  
if (front == null)  
    front = p  
else  
    rear->next = p;  
rear = p
```



الف

ب



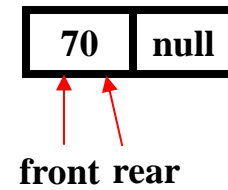


صفحه ۱۹

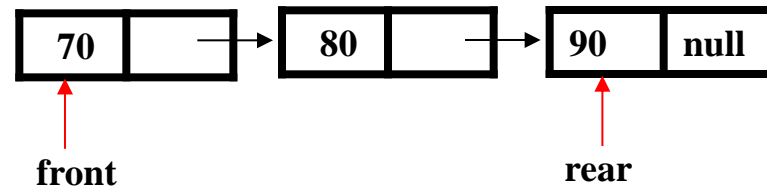
ساختمان داده ها و الگوریتم ها

عمل حذف از صف (delq):

الف



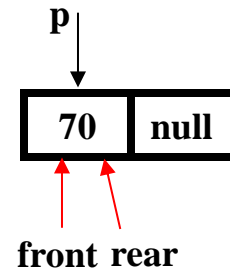
ب





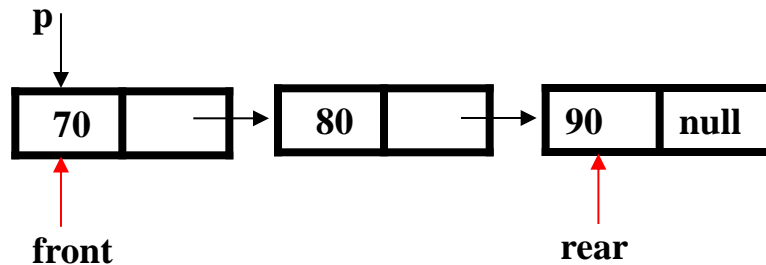
عمل حذف از صف (delq):

```
node *p;  
p = front;
```



الف

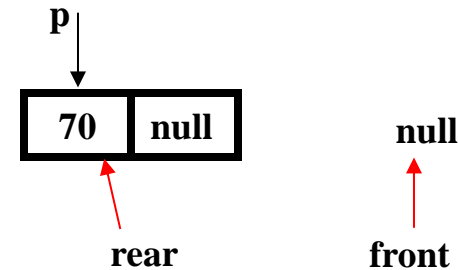
ب



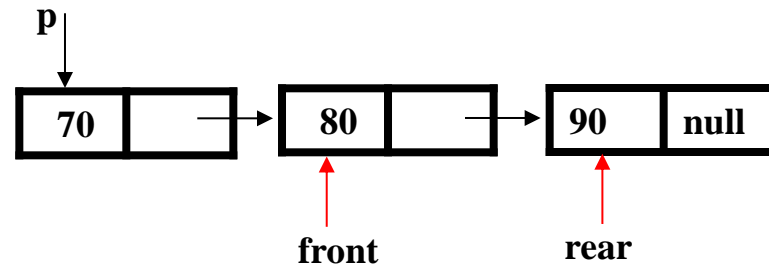


عمل حذف از صف (delq):

```
node *p;  
p = front;  
front = front->next;
```



الف



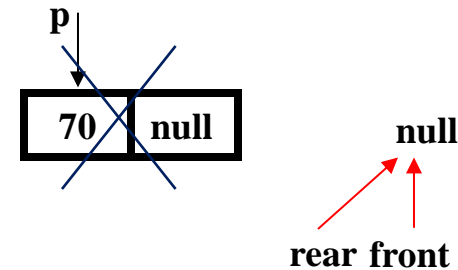
ب



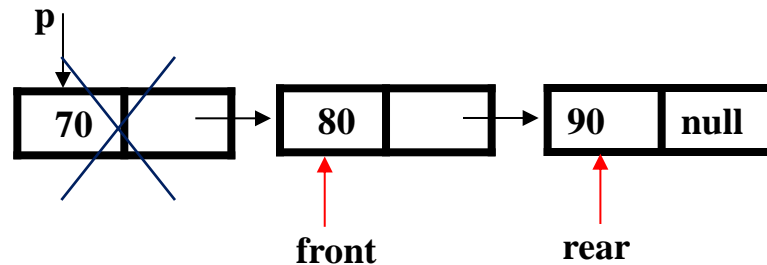
عمل حذف از صف (delq):

```
node *p;  
p = front;  
front = front->next;  
if (front == null)  
    rear = null;  
delete (p);
```

الف



ب





ساختمان داده ها و الگوریتم ها

صفحه ۲۳

تمرین ۱: توابع push و pop را طوری بنویسید که در هر بار عمل push دو عنصر اضافه و در هر بار عمل pop دو عنصر حذف شود؟

تمرین ۲: توابع addq و delq را طوری بنویسید که در هر بار عمل addq دو عنصر اضافه و در هر بار عمل delq دو عنصر حذف شود؟



پایان