

```

In [1]: 1 # A
2 class ArrayStack:
3     def __init__(self,size):
4         self.size=size
5         self.data=[0 for i in range(size)]
6         self.top=0
7     def isEmpty(self):
8         if self.top==0:
9             return True
10        else:
11            return False
12    def Push(self,value):
13        if self.top==self.size:
14            print("Stack Overflow !")
15        else:
16            self.data[self.top]=value
17            self.top+=1
18    def Pop (self):
19        if self.isEmpty():
20            print("Stack Underflow !")
21        else:
22            x=self.data[self.top-1]
23            self.top-=1
24            self.data[self.top]=0
25            return x
26    def Check(self):
27        if self.isEmpty:
28            True
29        else:
30            False
31    def Peek(self):
32        return self.data[self.top-1]
33    def Count(self):
34        return self.top
35    def Printt(self):
36        for i in self.data:
37            print(i)
38    ob=ArrayStack(3)
39    ob.Push(7)
40    ob.Push(6)
41    ob.Push(5)
42    ob.Push(77)
43    #ob.Pop()
44    print(ob.Peek())
45    print(ob.Count())
46    ob.Printt()

```

Stack Overflow !

5
3
7
6
5

In [4]:

```

1  # B
2  class ArrayQueue:
3      def __init__(self,size):
4          self.size=size
5          self.data=[0 for i in range(size)]
6          self.f=-1
7          self.r=0
8      def enqueue(self,value):
9          self.data[self.r]=value
10         self.r=(self.r+1)%self.size
11     def dequeue(self):
12         self.f=(self.f+1)%self.size
13         return self.data[self.f]
14     def isEmpty(self):
15         if self.f==-1 and self.r==0:
16             return True
17         else:
18             return False
19     def Count(self):
20         return len(self.data)
21     def Printt(self):
22         print(self.Count())
23 ob=ArrayQueue(4)
24 ob.enqueue(1)
25 ob.enqueue(2)
26 ob.enqueue(3)
27 ob.enqueue(4)
28 ob.enqueue(5)
29 print(ob.dequeue())
30 print(ob.data)
31 ob.Printt()

```

```

5
2
[5, 77, 3, 4]
4

```

In [6]:

```

1  # C
2  class ArrayStack:
3      def __init__(self,lst):
4          self.lst=lst
5          self.size=len(self.lst)
6          self.data=[0 for i in range(self.size)]
7          self.top=0
8      def StringExp(self):
9          for i in self.lst:
10             #print(self.data)
11             if i=="{" or i=="(" or i=="[" :
12                 self.Push(i)
13             if i=="}" or i==")" or i=="]" :
14                 self.Pop()
15             return self.isEmpty()
16     def isEmpty(self):
17         if self.top==0:
18             return True
19         else:
20             return False
21     def Push(self,value):
22         if self.top==self.size:
23             print("Stack Overflow !")
24         else:
25             self.data[self.top]=value
26             self.top+=1
27     def Pop(self):
28         if self.isEmpty():
29             print("Stack Underflow !")
30         else:
31             x=self.data[self.top]
32             self.top-=1
33             self.data[self.top]=0
34             return x
35 str1="{()}[()]{}"
36 ob=ArrayStack(str1)
37 ob.Pop()
38 print(ob.StringExp())

```

```

Stack Underflow !
True

```