السؤال 1

public interface Stack<T>{  
 void push(T element);  
   
 T pop();  
   
 T peek();  
   
 boolean isEmpty();  
   
 int size();  
   
 void clear();  
 }  
   
}

السؤال الثاني

private int self;  
def\_init\_(self);  
self.stack = []  
  
def is\_empty(self);  
return len(self.stack) ==0  
  
def push(self,item):  
self.stack.append(item)  
  
def pop(self):  
if self.is\_empty():  
return self.stack.pop().  
  
def peek(self):  
if self.is\_empty():  
return self.stack[-1]  
  
def size(self):  
return len(self.stack)

السؤال 3

public class m3 {  
  
  
 m2.def is\_empty(self);  
 return self.head is None  
   
 class def {}  
 def push(self,data):  
 new\_node = Node(data);  
   
 def pop(self):  
 return None;  
   
 def peek (slef:  
 if self.is\_empty():  
 return None;  
   
 def peek(self):  
 if self.is\_empty():  
 return None;  
 return self.head.data;

السؤال 4

public class m4 {  
  
 class def{}  
 def is\_empty(self):  
 return len (self.stack)==0  
 def size(self):  
 return len (self.stack);  
 def top (self):  
 if is\_empty(self):  
 return self.stack[-1]  
def push(self,element):  
 self.stack.append(element);  
   
 stack=stack()  
 print(stack.is\_empty());  
 print(stack.size())  
  
 stack.push(1)  
 stack.push(2)  
 stack.push(3)  
  
 print(stack.top())  
 print(stack.size())  
  
 stack.pop()  
 print(stack.top())  
 print(stack.size())

السؤال 5

public class m5 {  
 class def{}  
 def transfer(S,T):  
 while not S.is\_empty():  
 T.push(S.pop())  
}

السؤال 6

public class m6 {  
 class def{}  
   
 def remove\_elements(stack):  
 if not stack:  
 return  
 stack.pop()  
 remove\_elements(stack)  
 my\_stack =[1,2,3,4,5]  
 remove\_elements(my\_stack)  
 print(my\_stack)  
   
}

السؤال 7

1.قم بأنشاء مكدس فارغ الاحتفاظ بالمعاملات.

2.اقرأالتعبير من اليسار إلى اليمين رمزا وحدا في كل مرة.

3.إذا كان الرمز رقما أو متغيرا فأدفعه إلى المكدس.

4.إذا كان الرمزالمميزعامل التشغيل فقم بإخراج العنصرين العلويين من المكدس كعاملين.

5.قم بإجراء العملية على المعاملات باستخدام عامل التشغيل.

6. ادفع نتيجة العلمية مرة اخري إلى المكدس.

7.كرر الخطوات من 3 إلى 6 حتى تتم المعالجة كافة الرموز المميزة في التعبير.

8.بعدمعالجة جميع الرموز ستكون النتيجة النهائية هي العنصر الوحيد المتبقي في المكدس.

الخوارزمية

1.ابدا بمكدس فارغ.

2.اقرا الرمز الاول5.

. ادفعه على المكدس5.

3.اقرا الرمز التالي 2.

ادفعه على المكدس 2و5.

4.اقرا الرمز التالي"+".

قم بإخراج العنصرين العلويين من المكدس 2و5.

إجراء عملية الجمع2+5=7.

ادفع النتيجة إلى المكدس 7.

السؤال 8

public class m8 {  
 implement cloneable{  
 private T[] array;  
 private int size;  
 private static final int  
   
 DEFAULT\_CAPACITY =10;  
 public ArrayStaack(){  
 array =(T[]) new  
 size =0;  
 super.clone();  
 array.clone();  
 }  
 }  
  
}

السؤال 9

def eveluate(self,expressions);  
 for token.isdigit()  
 def\_perform\_operation  
   
 r def push(self,item):  
 self.stack.append(item)  
  
 m2.def pop(self):  
 if self.is\_empty():  
 return self.stack.pop().  
  
 m2.def peek(self):  
 if self.is\_empty():  
 return self.stack[-1]  
  
 m2.def size(self):  
}