المصفوفات و السجلات و الخوارزميات في باسكال.

الصف الثالث الثانوي

1. المصفوفات

**1/ تتفق لغات البرمجة في أن لا بد من تعريف المصفوفة في بداية البرنامج في جزء التعريفات بعد المتغيرات البسيطة تحت الكلمة المحجوزة Var . و يكون كالآتــــي :**

**( الاسم : كلمة مصفوفة [ عدد أبعادها و طول كل بعد ] نوع بياناتها .**

نوع بياناتها of ] أطوال إبعاد المصفوفة [ array:أسم المصفوفة

Var

St\_ scores : array [ 1.. 20 , 1.. 5 ] of integer ;

حيث St-score اسم المصفوفة

1-20 البعد الأول و طوله 20

1-5 البعد الثاني و طوله 5

Of integer نوع البيانات ( رقمية )

(مصفوفة ذات بعد واحد)Students :array[1—30]of Integer;

(ذات بعدين (30 × 9) St-sud: array [1---40,1—9]ofCHar;

ب)**إدخال قيم المصفوفة:~**

St-score : array [1—50]of integer;

(إسناد القيمة 100 إلي الصف رقم 10 في المصفوفة ) St-score [10]=100];

(من لوحة المفاتيح ضع قيمة في الصف رقم 10) Readln(St-score[10]);

1. **السجلات :~**

لا يتعامل باسكال مع السجلات مباشرة لكنها تتعامل معها كمجموعة من المصفوفات ( كل حقل في السجل يمثل مصفوفة)

لدينا السجل sch به الحقول num,nam, sco, tot , per يتم تعريفه كالآتي :~

Type

Sch=Record

Num:Integer ;

Nam:string ;

Sco: integer ;

Per :Boolean ; School : array [1—100] of Sch;

End ;

Var

School: array [ 1 – 100 ] of Sch ;

1. \*\*\* المكدسات :

**\*\*\*خوارزمية الإضافة لمكدسة :~**

1/ هل الموقع الأعلى أصبح يساوي أو اكبر من حجم المكدسة (هل المكدسة ممتلئة ) ؟ نعم ، لا يمكن الإضافة مع رسالة (Over flow)وتنتهي ألخوارزميه ،إما لا تستمر في الإضافة

2/الموقع الاعلي =الأعلى +1.(حرك الموقع الاعلي خطوة للأعلى )

3/ المصفوفة ( الاعلي ) = العنصر المضاف

***\*\*\* برنامج باسكال لخوارزمية الإضافة في مكدسة :~***

Const

Max = 100 ;

Var

Top : integer ;

Newelement : char ;

Stack : array [ 1 .. Max ] of char ;

Begin

Readln ( Newelement ) ;

If Top = Max Then

Writeln ( ' over flow ' ) ;

Else

Begin

Top := Top + 1 ;

Stack [ Top ] := Newelement ;

End ;

Writeln ( Newelement ) ;

End .

***\*\*\*خوارزميه الحذف من مكدسه :~***

1/هل الموقع الأعلى =صفر (هل المكدس فارغ )؟ نعم ، لا يمكن الحذف وتنتهي ألخوارزميه .لا تستمر في الحذف .

2/العنصر المحذوف = المصفوفة ( الموقع الأعلى )، (حذف العنصر الذي في الموقع الاعلي للمصفوفة )

3/الموقع الاعلي = الاعلي -1 0

Const

Max = 100 ;

Var

Top : integer ;

Delelement : char ;

Stack : array [ 1 .. Max ] of char ;

Begin

Readln ( Top ) ;

If Top = 0 Then

Writeln ( 'under flow ' ) ;

Else

Begin

Delelement:= Stack [ Top ] ;

Top := Top – 1 ;

End ;

End .

1. **\*\* وصف خوارزمية البحث المتتالي**
2. أدخل عدد المفاتيح (ن) و مصفوفة المفاتيح .
3. أدخل المفتاح المطلوب .
4. أجعل عداد المفاتيح في البدء = صفر .
5. العداد = العداد + 1 .
6. أقرأ المفتاح الذي يشير إليه المؤشر .
7. هل تطابق مع المفتاح المطلوب ؟ نعم ، أذهب إلي 8 .
8. هل العداد أكبر من (ن) ؟ " لا " ، أذهب إلي 4 ، " نعم " ، أذهب إلي 10 .
9. أكتب وجد المفتاح في مؤشر العداد .
10. أذهب إلي 11
11. أكتب لم يوجد المفتاح .
12. النهاية .

**\*\*برنامج بلغة باسكال لتنفيذ خوارزمية البحث المتتالي :-**

**Const**

N = 50 ;

Var

I, Key : Integer ;

Found : Boolean ;

Keys : array [ 1.. N ] of integer ;

Begin

For I := 1 To N Do

Read ( Keys [ I ] ) ;

I := 0 ;

Found := False ;

Read ( Key ) ;

Repeat

I := I + 1 ;

If Keys [ I ] = Key Then

Found := True ;

Until

I > N OR Found = True ;

If I > N Then

Write ( ' Key not found ' ) ;

Else

Write ( ' Key found in ' , I ) ;

End .

1. **وصف خوارزمية البحث الثنائي**
2. أدخل مصفوفة المفاتيح و المفتاح المطلوب .
3. هل المؤشر الأول = المؤشر الآخير ؟ نعم ، أذهب إلي 6 .
4. أحسب مؤشر الوسط = (الأول + الآخير ) /2 .
5. هل مفتاح مؤشر الوسط = المفتاح المطلوب ؟ نعم ، أذهب إلي 8 .
6. إذا كان مفتاح مؤشر الوسط أكبر من المفتاح المطلوب ، أجعله الآخير و إلا أجعله الأول .
7. أكتب لا يوجد المفتاح .
8. أذهب إلي 9 .
9. أكتب مؤشر الوسط .
10. النهاية .

**\*\* برنامج باسكال لخوارزمية البحث الثنائي :-**

Const

N = 50 ;

Var

I, Key , First , middle , last : Integer ;

Found : Boolean ;

Keys : array [ 1.. N ] of integer ;

Begin

For I := 1 To N Do

Read ( Keys [ I ] ) ;

Read ( Key ) ;

Found := False ;

First := 1 ;

Last := N ;

Repeat

Middle := ( first + last ) Div 2 ;

If Keys [ middle ] > Key Then

Begin

Last := middle ;

If Keys [ middle ] < Key Then

First := middle ;

Else

Found := True ;

End ;

Until

Found OR last <= first

If found Then

Write ( ' Key found ' , middle ) ;

Else

Write ( ' Key not found ' ) ;

End .

1. **التشفير ( يوليوس قيصر ) :-**

الحروف الأخيرة في جدول الحروف ( عددها = مفتاح الشفرة ) عندما تشفر بإضافة المفتاح تتعدي حروف الجدول ، و تعالج هذه المشكلة بطرح العدد الكلي للحروف من الرقم الناتج بإضافة المفتاح . و برمجيا باستخدام الدالة Mod

31 mod 28 = 3 -- 30 mod 28 = 2 – 29 mod 28 = 1 .

**برنامج بلغة باسكال للتشفير باستخدام يوليوس قيصر :-**

Program Kaiser ;

Const

No\_of\_char = 28 ;

Var

I : integer ;

Letters : array [ 1 .. No\_of\_char ] of char ;

Begin

For I := 1 To No\_of\_char Do

End .

\*\* لفك الشفرة نستخدم نفس البرنامج أعلاه مع إجراء التغيير التالي :

Letters[ I ] := char (( I + 3 )mod No\_of\_char) ;

Letters[ I ] := char (( I - 3 )mod No\_of\_char) ;

عثمان يوسف توتو

0122714573