العمل التطبيقي 2:

الهدف: التمرن على كتابة بعض الخوارزميات مع مراعاة التدرج في التعقيد وفي استعمال التركيبات المختلفة.

1- خوارزميات حسابية بسيطة:

1- اكتب خوارزم لحساب مجموع عددين وطباعة النتيجة على الشاشة.
2- اكتب خوارزم لحساب محيط ومساحة مستطيل.
3- اكتب خوارزم لحساب محيط ومساحة دائرة.
4- اكتب خوارزم لتحويل مسافة من الكيلومتر إلى الميل وإظهار النتيجة (1 ميل = 1.609 كلم)
5- اكتب خوارزم لحساب محيط ومساحة مثلث من خلال معرفة أطوال أضلاعه الثلاثة. لحساب المساحة نعتمد على صيغة هيرون (Heron) والمعرفة بـ: $S = \sqrt{p(p-a)(p-b)(p-c)}$ حيث p هو نصف المحيط و a، b، c هي أطوال الأضلاع الثلاثة.
6- اكتب خوارزم لتبديل قيمتي عددين A وB فيما بينهما.

2- خوارزميات بسيطة مع استعمال تركيبات الاختيار البسيط "إذا" وتركيبة الاختيار المتعدد "حسب":

1- اكتب خوارزم لقراءة عددين ثم إظهار العدد الأكبر.
2- اكتب خوارزم لقراءة عددين A وB ثم تبديل قيمتيهما إذا كانت قيمة A أصغر من قيمة B .
3- اكتب خوارزم لقراءة معدل طالب ثم إظهار النص "ناجح" أو "راسب" على الشاشة.
4- عدل الخوارزم السابق بإضافة النص "علامة خاطئة" إذا كانت علامة المعدل المدخلة أصغر من 0 أو أكبر من 20.
5- اكتب خوارزم لطباعة اسم اليوم الموافق لرقم محصور بين 0 و6 (0: "السبت"، 1: "الأحد"،...، 6: "الجمعة") بالاعتماد على التركيبة "إذا" ثم بالاعتماد على التركيبة "حسب".

3- خوارزميات باستعمال تركيبات التكرار:

- اكتب خوارزم لطباعة الأعداد من 1 إلى N ثم إظهار المجموع.
- اكتب خوارزم لقراءة علامة الامتحان في مادة الاعلام الآلي مع إجبار المستعمل على إدخال قيمة صحيحة (محصورة بين 0 و20).
- اكتب خوارزم للبحث عن عدد إن كان موجودا داخل قائمة غير مرتبة من الأعداد (أو جدول).
- اكتب خوارزم للبحث عن عدد داخل قائمة (جدول) مرتبة ترتيبا تصاعديا بالاعتماد على طريقة البحث "ثنائي التفرع" (Dichotomic Search) (Recherche dychotomique).
- اكتب خوارزم لحساب عاملي عدد (N !).
- اكتب خوارزم لمعرفة إن كان العدد المدخل أوليا أم لا.
- اكتب خوارزم لقراءة عناصر جدول ثم ترتيبها ترتيبا تصاعديا.
- اكتب خوارزم لحساب قيمة تقريبية للدالة الأسية للعدد x (e^x) بالاعتماد على نشر تايلور المعرف بـ:

$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!}$$

(نتوقف عن الحساب عند أول حد أصغر من قيمة ε (0.001 = ε مثلا)

الحل النموذجي

1- خوارزميات حسابية بسيطة:

1- خوارزم حساب مجموع عددين وطباعة النتيجة على الشاشة (الحل موجود في الدرس) :

خوارزم جمع

A، B، S: أعداد صحيحة

بداية

اقرأ (A ، B)

S = A + B

اكتب (S)

نهاية

Algorithm Sum

A , B, S : integer

Begin

Read(A, B)

S=A + B

Write(S)

End

Algorithme Somme

A , B, S : entier

Début

Lire(A, B)

S=A + B

Ecrire(S)

Fin

2- خوارزم لحساب محيط ومساحة مستطيل:

خوارزم مستطيل

A، B، S، P: أعداد صحيحة

بداية

اقرأ (A ، B)

P ← 2*(A+B)

S ← A * B

اكتب (P ، S)

نهاية

Algorithm Sum

A, B, P, S: integer

Begin

Read(A, B)

P←2*(A+B)

S←A * B

Write(P, S)

End

Algorithme Somme

A , B, P S : entier

Début

Lire(A, B)

P←2*(A+B)

S←A * B

Ecrire(P, S)

Fin

3- خوارزم لحساب محيط ومساحة دائرة (تم حله في حصة المحاضرة).

4- خوارزم لتحويل مسافة من الكيلومتر إلى الميل وإظهار النتيجة (1 ميل = 1.609 كلم) :

خوارزم تحويل

K، M: أعداد حقيقية

بداية

اقرأ (K)

M ← K / 1.609

اكتب (M)

نهاية

Algorithm Converting

K, M: integer

Begin

Read(K)

M←K/1.609

Write(M)

End

Algorithme Conversion

K, M: entier

Début

Lire(K)

M←K/1.609

Ecrire(M)

Fin

5- خوارزم لحساب محيط ومساحة مثلث من خلال معرفة أطوال أضلاعه الثلاثة:

خوارزم مثلث
A، B، C،P،S: أعداد حقيقية
بداية
اقرأ (A، B،C)
P ← A+B+C
اكتب (P)
P←P/2
S←sqrt(P*(P-A)*(P-B)*(P-C))
اكتب(S)
نهاية

```
Algorithm Triangle
    A,B,C,P,S: integer
Begin
    Read(A,B,C)
    P←A+B+C
    Ecrire(P)
    P←P/2
    S←sqrt(P*(P-A)*(P-B)*(P-C))
    Write(S)
End
```

6- خوارزم لتبديل قيمتي عددين A وB فيما بينهما:

خوارزم تبديل
A، B، C: أعداد حقيقية
بداية
اقرأ (A، B)
C ← A
A←B
B←C
اكتب (A،B)
نهاية

```
Algorithm Permutation
    A,B,C: integer
Begin
    Read(A,B)
    C←B
    B←A
    A←C
    Write(A,B)
End
```

2- خوارزميات بسيطة مع استعمال تركيبات الاختيار البسيط "إذا" وتركيبة الاختيار المتعدد "حسب":

1- خوارزم لقراءة عددين ثم إظهار العدد الأكبر:

خوارزم قراءة
A، B: أعداد حقيقية
بداية
اقرأ (A، B)
إذا A >= B إذن
اكتب(A)
وإلا
اكتب (B)
نهاية إذا
نهاية

```
Algorithm R1
    A,B: real
Begin
    Read(A,B)
    If A >= B Then
        Write(A)
    Else
        Write(B)
    End if
End
```

2- خوارزم لقراءة عددين A وB ثم تبديل قيمتيهما إذا كانت قيمة A أصغر من قيمة B :

خوارزم تبديل2
A، B، C: أعداد حقيقية
بداية
اقرأ (A، B)
إذا A > B إذن
C ← A
A←B
B←C
نهاية إذا
اكتب (A،B)
نهاية

Algorithm Permutation2
A,B,C: integer
Begin
Read(A,B)
If A<B then
C←B
B←A
A←C
End if
Write(A,B)
End

3- خوارزم لقراءة معدل طالب ثم إظهار النص "ناجح" أو "راسب" على الشاشة:

خوارزم نتيجة
A: عدد حقيقي
بداية
اقرأ (A)
إذا A >= 10 إذن
اكتب("ناجح")
وإلا
اكتب("راسب")
نهاية إذا
نهاية

Algorithm Obs
A: real
Begin
Read(A)
If A>=10 then
Write("Pass")
Else
Write("Fail")
End if
End

4- تعديل الخوارزم السابق بإضافة النص "علامة خاطئة" إذا كانت علامة المعدل المدخلة أصغر من 0 أو أكبر من 20:

يتم ذلك بإدراج الشرط التالي مباشرة بعد قراءة المعدل:

إذا (A < 0) أو (A > 20) إذن
اكتب ("علامة خاطئة")
وإلا
إذا A > = 10 إذن
...

5- خوارزم لطباعة اسم اليوم الموافق لرقم محصوربين 0 و6 (0: "السبت"، 1: "الأحد"،... 6: "الجمعة") بالاعتماد على التركيبة "إذا" ثم بالاعتماد على التركيبة "حسب":

```
خوارزم يوم
A: عدد صحيح
بداية
    اقرأ (A)
    إذا A = 0 إذن
        اكتب("السبت")
    وإلا
        إذا A = 1 إذن
            اكتب ("الأحد")
        وإلا
            إذا A = 2. إذن
            ...

    نهاية إذا
```

باستعمال حسب (switch)

```
خوارزم يوم
A: عدد صحيح
بداية
    اقرأ (A)
    حسب A
        حالة 0 : اكتب("السبت")
        حالة 1 : اكتب("الأحد")
        حالة 2: اكتب("الإثنين")
        حالة 3 : اكتب("الثلاثاء")
        حالة 4: اكتب("الأربعاء")
        حالة 5 : اكتب("الخميس")
        حالة 6 : اكتب("الجمعة")
        وإلا : اكتب("خطأ")
    نهاية حسب
نهاية
```

```
Algorithm R1
    A: integer
Begin
    Read(A)
    Switch A
        Case 0: Write("Saturday")
        Case 1: Write("Sanday")
        Case 2: Write("Monday")
        Case 3: Write("Tuesday")
        Case 4: Write("Wednesday")
        Case 5: Write("Thirsday")
        Case 6: Write("Friday")
    Else: Write("Error")
    End switch
End
```

3- خوارزميات باستعمال تركيبات التكرار:

1- خوارزم لطباعة الأعداد من 1 إلى N ثم إظهار المجموع:

(تم حله في حصة المحاضرة).

2- خوارزم لقراءة علامة الامتحان في مادة الاعلام الآلي مع إجبار المستعمل على إدخال قيمة صحيحة (محصورة بين 0 و20).

(تم حله في حصة المحاضرة).

3- اكتب خوارزم للبحث عن عدد إن كان موجودا داخل قائمة غير مرتبة من الأعداد (أو جدول).

(سيتم حله في حصة المحاضرة إن شاء الله).

4- اكتب خوارزم للبحث عن عدد داخل قائمة (جدول) مرتبة ترتيبا تصاعديا بالاعتماد على طريقة البحث "ثنائي التفرع" (Dichotomic Search) (Recherche dychotomique).

(سيتم حله في حصة المحاضرة إن شاء الله).

5- اكتب خوارزم لحساب عاملي عدد (N !).

```
خوارزم عاملي
    N ،F، i : أعداد صحيحة
بداية
    اقرأ (N )
    F←1
    من أجل i = 2 إلى N افعل
        F ← F * i
    نهاية من أجل
    اكتب (F)
نهاية
```

```
Algorithm Factorial
    N, i, F: integer
Begin
    Read(N)
    F←1
    For i =1 : N do
        F←F * i
    End for
        Write(F)
End
```

6- خوارزم لمعرفة إن كان العدد المدخل أوليا أم لا.

خوارزم أولي

N، i : أعداد صحيحة

R : عدد منطقي

بداية

اقرأ (N)

i←2

R←صحيح

طالما (i <= √N) و (R=صحيح) افعل

إذا N mod i = 0 إذن

R←خطأ

وإلا

i ← i +1

نهاية إذا

نهاية طالما

إذا (R=صحيح) إذن

اكتب ("هذا العدد أولي")

وإلا

اكتب ("هذا العدد ليس أولي")

نهاية إذا

نهاية

```
Algorithm Factorial
    N, i : integer
    R: boolean
Begin
    Read(N)
    i←2
    R← true
    while (i <= sqrt(N)) And(R = True) do
        If N mod i = 0 then
            R←False
        Else
            i← i +1
        End if
    End while
    If R then
            Write("This nubmer is prime")
    Else
            Write("This nubmer is not prime")
    End if
End
```

يمكن استعمال " sqrt(N)" للتعبير عن الجذر لتربيعي.

7- خوارزم لقراءة عناصر جدول ثم ترتيبها ترتيبا تصاعديا.

خوارزم جدول

T: جدول [1:100]من الأعداد صحيحة

i،N : عدد صحيح

بداية

اقرأ (N)

من أجل i = 1 إلى N افعل

اقرأ(T[i])

نهاية من أجل

نهاية

```
Algorithm table
    N,i: integer
    T: array[1:100]of integer
Begin
    Read(N)
    For i =1 : N do
        Read(T[i])
    End for

End
```

N: يمثل عدد العناصر الموجودة فعليا في الجدول، وهو يجب أن يكون أصغر من 100 لأن الطول الأقصى للجدول المصرح به هنا هو 100.

// الترتيب (طريقة الفقاعة Bubble sort)

المبدأ: نقارن كل عنصر مع العنصر الذي يليه إن كان أكبر منه نقوم بعملية التبديل بينهما حتى نصل إلى آخر عنصر. نتيجة هذه المرحلة هو أن العنصر الأكبر يكون قد وضع في آخر خانة (أي رتبنا العنصر الأكبر) نعيد الكرة على العناصر من الأول إلى العنصر ما قبل الأخير وهكذا حتى الوصول إلى ترتيب كل العناصر. ميزة هذه التقنية هي أن الجدول يرتب ابتداء من نهاية الجدول ووصولا إلى بدايته.

خوارزم ترتيب-جدول

T: جدول [1:100]من الأعداد صحيحة

i،M،N،j : عدد صحيح

بداية

من أجل i = 1 إلى N-1 افعل

من أجل j = 1 إلى N-1-i افعل

إذا T[j] > T[j+1] إذن

M←T[j]

T[j]←T[j+1]

T[j+1]←M

نهاية إذا

نهاية من أجل

نهاية من أجل

نهاية

```
Algorithm table-sort
    N,i,j; M: integer
    T: array[1:100]of integer
Begin
    For i =1 : N-1 do
        For j =1 : N-1-i do
          If T[j] > T[j+1]
              M←T[j]
              T[j]←[j+1]
              T[j+1]← M
          End if
        End for
    End for
End
```

8- خوارزم لحساب قيمة تقريبية للدالة الأسية للعدد x (e^x) بالاعتماد على نشر تايلور:

خوارزم أسية

ثابت : Epsilon = 0.001

F،i : عدد صحيح

X،E،P،T: عدد حقيقي

بداية

اقرأ(X)

T←1 ، F←1 ، P←1 ،i←1 ، E←1

طالما T >= Epsilon افعل

P← P * X

F←F * i

T←P/F

E←E + T

i ← i + 1

نهاية طالما

اكتب(E)

نهاية

```
Algorithm Exponential
    Constant :  Epsilon=0.001
    F, i:integer
    E,X,P,T: real
Begin
    Read(X)
    E←1 ; i←1 ;P←1 ;F←1 ;T←1
    While T>= Epsilon
        P←P*X
        F←F*i
        T←P/F
        E←E+T
        i ← i+1
    End while
    Write(E)
End
```

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \ldots \frac{x^n}{n!} + \ldots$$

T: يمثل الحد الذي يحسب في كل مرة، يتكون من الكسر $\frac{P}{F}$ ، وهو الذي يحدد متى تتوقف عملية التكرار (بمجرد أن نصل إلى حد أصغر من 0.001).

P: لحساب الأس (x^n) ومادام X عدد حقيقي فإن P يكون حقيقيا.

F: لحساب العاملي $(n!)$ وهو عدد صحيح.

E: يمثل النتيجة (مجموع الحدود).

يمكن إضافة التمثيل عن طريق المخطط الانسيابي: