



مقدمة في البرمجة Introduction To Programming

IPG101

الفصل الثالث تمارين عملية على تصميم البرنامج Practice On Program Design

الكلمات المفتاحية

خوارزمية، برنامج، خوارزمية حسابية، خوارزمية غير حسابية، خوارزمية تسلسلية، خوارزمية غير تسلسلية، قرار، تكرار، مخطط تدفقي، شبه الشيفرة.

ملخص الفصل

يتضمن هذا الفصل مجموعة من التمارين التطبيقية على تصميم البرنامج وذلك بهدف تعزيز المفاهيم النظرية التي تلقاها خلال الفصل السابق واكتساب المهارات العملية في التعامل مع المسائل المطروحة بدءاً بعملية تحليل المسألة وتوصيف مكوناتها وانتهاء بتصميم خوارزمية الحل المناسبة والتعبير عنها باستخدام أساليب المخططات التدفقية وشبه الشيفرة، حيث تنتوع المسائل المطروحة مابين خوارزميات تسلسلية وخوارزميات قرار وتكرار بما يمكنه من التعامل مع طيف أوسع من أنواع المسائل وصياغة الحلول المناسبة لها.

أهداف الفصل

بنهاية هذا الفصل سيكون الطالب قادراً على:

- تعزيز مهاراته في تحليل المسائل وتصميم الحلول.
- التعبير عن حلول المسائل المختلفة على شكل خوارزميات.
- تصميم خوارزميات القرار والتكرار.
- كتابة شبه الشيفرة الممثل لخطوات خوارزمية ما.
- رسم المخططات التدفقية للخوارزميات المختلفة.

محتويات الفصل

1. مقدمة.
2. التمرين الأول : مسألة جداء عددين.
3. التمرين الثاني: مسألة قسمة عددين.
4. التمرين الثالث: مسألة محيط ومساحة الدائرة.
5. التمرين الرابع: مسألة العدد الأكبر.
6. التمرين الخامس: مسألة القاسم المشترك الأكبر لعددين.
7. التمرين السادس: مسألة جذور معادلة من الدرجة الثانية.
8. التمرين السابع: مسألة الأعداد المحصورة بين عددين.
9. التمرين الثامن: مسألة حساب العامل.

1- مقدمة

قبل البدء بعرض التمارين المختلفة ومناقشة حلولها، لابد من تقديم موجز بأهم الأفكار المتعلقة بتصميم البرنامج والتعبير عن الخوارزميات.

الخوارزمية هي مجموعة متتالية ومنتهية من الخطوات التي تعبر عن حل مسألة ما وبالتالي يتضمن تصميم خوارزمية حل المسألة أمرين أساسيين وهما تحديد الأفعال الواجب تنفيذها والترتيب الواجب إتباعه لدى تنفيذ الأفعال السابقة للوصول إلى الحل.

الخوارزمية غير الحسابية هي الخوارزمية التي لا تتضمن التعامل مع مقادير حسابية ولا عمليات حسابية أو رياضية ولا تعيد نتائج رقمية.

الخوارزمية الحسابية هي الخوارزمية التي تتعامل مع المقادير الرياضية.

يمكن التعبير عن الخوارزميات بأساليب عدة مختلفة، منها اللغات الطبيعية natural languages، شبه الشيفرة pseudocode، المخططات التدفقية flowcharts، ولغات البرمجة programming languages.




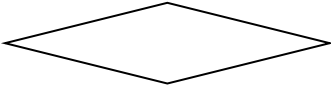
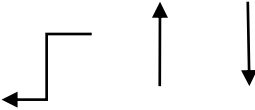
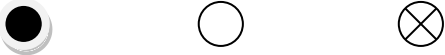
تقوم طريقة شبه الشيفرة على استخدام عبارات خاصة في التعبير عن خطوات الخوارزمية قريبة جداً من العبارات المستخدمة في لغات البرمجة إلا أنها مستقلة عن أي لغة برمجة. واستخدام صياغات رياضية شبه برمجية للعمليات وبنى تخزين القيم.

ملخص الرموز المستخدمة:

العملية	أوامر شبه الشيفرة
البداية والنهاية	START, BEGIN, STOP, END
الإدخال	INPUT, READ, GET
الإخراج	PRINT, DISPLAY, SHOW, PROMPT, WRITE, OUTPUT
التهيئة والإسناد	SET, INIT
العملية الحسابية	COMPUTE, CALCULATE, DETERMINE
القرار أو الشرط	TEST, IF/ELSE
الحلقة أو التكرار	WHILE, FOR, REPEAT, LOOP

المخطط التدفقي flowchart هو نوع من المخططات (الرسومية أو الرمزية) التي تعبر عن خوارزمية أو عملية ما. حيث يتم تمثيل كل خطوة في الخوارزمية بواسطة رمز مختلف يتضمن وصفاً مبسطاً لهذه الخطوة.

ملخص الرموز والأشكال المستخدمة:

العملية	الرمز التخطيطي المستخدم
البداية والنهاية	
الإدخال والإخراج	
العملية الحسابية والإسناد	
الاختبار أو القرار	
مسار التدفق	
نقاط التوصيل والربط	
إضافة إلى طيف واسع من الرموز الأخرى	

قبل البدء بكتابة أي خوارزمية، تأكد من أن المهمة محددة تمامًا وبالتالي لابد من البدء بتحليل المسألة عبر طرح مجموعة من الأسئلة ومحاولة الإجابة عنها بشكل واضح ودقيق.

أسئلة يجب طرحها:

- ما هي البيانات المعروفة قبل تشغيل البرنامج؟
- ما هي البيانات التي يجب إدخالها من قبل المستخدم؟
- ما هي الحسابات التي سيتم إجراؤها على البيانات؟
- ما هي البيانات التي سيتم إخراجها (المعروضة) للمستخدم؟

2- التمرين الأول: مسألة جداء عددين

المطلوب تصميم خوارزمية الحل لمسألة حساب ناتج ضرب عددين وإعطاء الناتج.

تحليل المسألة:

لحساب جداء عددين، لابد من أن يتوفر لدينا العددين المراد حساب جداولهما وبالتالي سيتم تقديم هذين العددين كمدخلات للمسألة المطروحة، ومن ثم إجراء عملية حساب ناتج الضرب وتقديمه كخرج للمسألة. لا تتضمن هذه المسألة أي قيم ثابتة أو مسبقة التعريف.

نلخص ماسبق كما يلي:

- القيم المعروفة

- لا يوجد.

- المدخلات

- العدد الأول.

- العدد الثاني

- الحسابات

- الناتج = العدد الأول x العدد الثاني

- المخرجات

- الناتج.

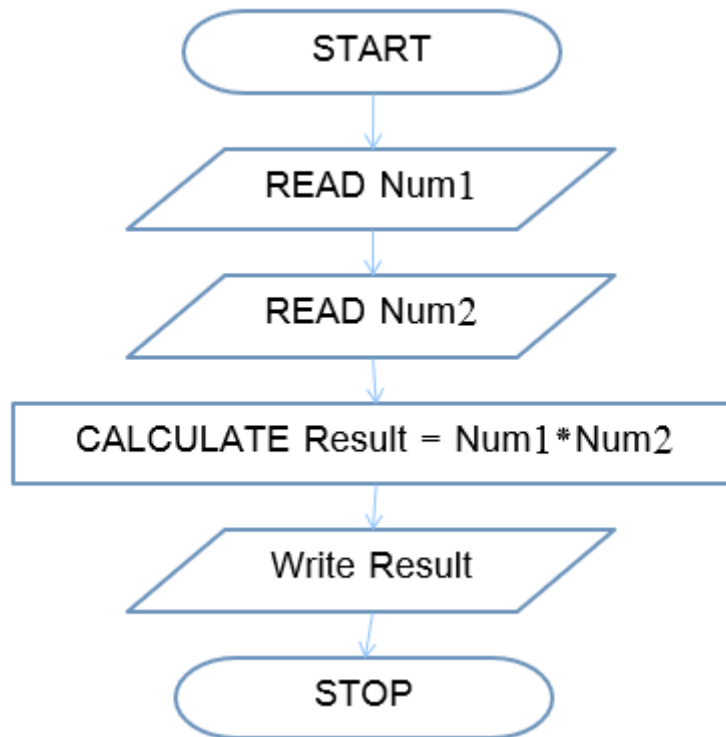
الحل باستخدام شبه الشيفرة:

1. **START**
2. **WRITE "Insert Num1"**
3. **READ Num1**
4. **WRITE "Insert Num2"**
5. **READ Num2**
6. **CALCULATE Result = Num1 x Num2**
7. **WRITE Result**
8. **STOP**

ملاحظات:

- ✓ إظهار الرسائل التوجيهية ليس جزءاً إلزامياً من الحل ولكنها مستحبة في الممارسات البرمجية عموماً لكننا سنتوقف عن إظهارها في التمارين اللاحقة من أجل الاختصار.
- ✓ من الممارسات الجيدة ترقيم الخطوات في شبه الشيفرة للدلالة على ترتيب التنفيذ وتتولى الأسهم في المخططات التدفقية هذه المهمة.
- ✓ يعتمد عادة بدء الخوارزمية بالكلمة **START** وإنهاؤها بالكلمة **STOP** (أو **BEGIN** ، **END**).

الحل باستخدام المخططات التدفقية:



مابعد الحل:

ربما يكون من المفيد أن تتعلم مهارة النمذجة الرياضية لأي مسألة مطروحة، فإذا أمعنا النظر في هذه المسألة، ندرك أنها قد طرحت بأبسط وأعم أشكالها، لكن هذا لا يعني أنه ليس هناك طيف واسع من المسائل التي يمكن أن تطرح وبسيناريوهات متعددة ولا يتجاوز تصميم حلها التصميم المطروح لهذه المسألة، فيما يلي بعض من الأمثلة:

- ✓ مسألة حساب مساحة قطعة أرض مستطيلة الشكل.
- ✓ مسألة حساب المبلغ المتوجب دفعه من مشتري في متجر قام بشراء عدد معين من القطع من سلعة ما.
- ✓ مسألة حساب مقدار الأرباح التي يجنيها التاجر لدى بيع سلعة معينة مع حساب الضريبة المتوجبة عن هذه الأرباح.
- ✓ حساب فرق الكمون الكهربائي على طرفي مقاومة لدى مرور تيار كهربائي عبرها.
- ✓ إلخ من طيف المسائل ذات البعد الرياضي أو المحاسبي المالي أو الفيزيائي.

3- التمرين الثاني: مسألة قسمة عددين

المطلوب تصميم خوارزمية الحل لمسألة حساب ناتج قسمة عددين وإعطاء الناتج.

تحليل المسألة:

لحساب جداء عددين، لابد من أن يتوفر لدينا العددين المراد حساب جداولهما وبالتالي سيتم تقديم هذين العددين كمدخلات للمسألة المطروحة مع الانتباه إلى أن العدد المقسوم عليه لا يمكن أن يكون صفراً، ومن ثم إجراء عملية حساب ناتج القسمة وتقديمه كخرج للمسألة.

نلخص ماسبق كما يلي:

- القيم المعروفة
 - لا يجوز أن يكون المقسوم عليه صفراً.
- المدخلات
 - العدد الأول.
 - العدد الثاني
- الحسابات
 - الناتج = العدد الأول ÷ العدد الثاني
- المخرجات
 - الناتج.

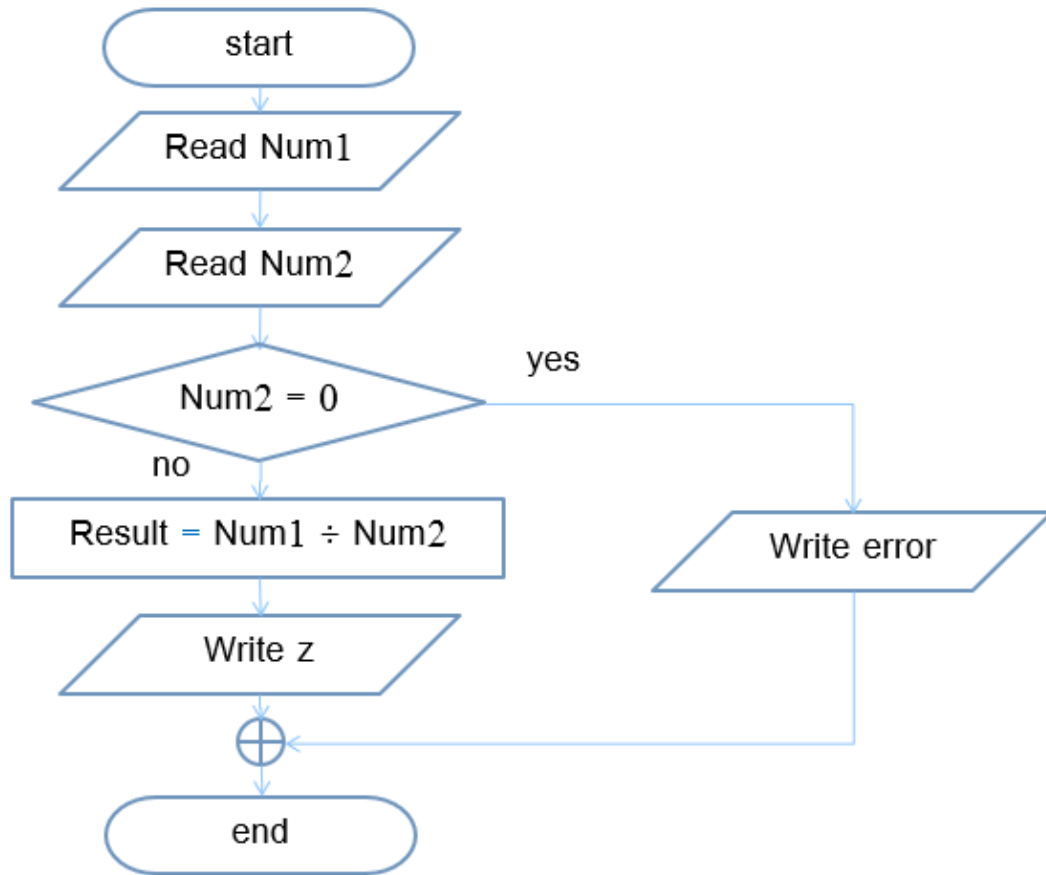
الحل باستخدام شبه الشيفرة:

```
1- START
2- READ Num1
3- READ Num2
4- IF Num2=0 THEN
    1- CALCULATE Result = Num1 x Num2
ELSE
    1- WRITE "ERROR DIVIDE BY ZERO"
5- WRITE Result
6- STOP
```

ملاحظات:

✓ من شروط الخوارزمية أن تراعي جميع الاحتمالات ولذا كان لابد من نقاش حالة القسمة على صفر.

الحل باستخدام المخططات التدفقية:



ما بعد الحل:

حاول الاستفادة من هذه الخوارزمية في حل المسألة التالية:

✓ مسألة حساب سرعة عربة بمعرفة المسافة التي قطعها خلال مدة من الزمن.

1- قارن بين الحلين، هل وجدت اختلافاً في الحل رغم اختلاف نص المسألة المطروح؟

2- إقترح مسائل أخرى ذات بعد فيزيائي أو رياضي أو محاسبي تملك نفس خوارزمية الحل السابقة.

4- التمرين الثالث: مسألة محيط ومساحة الدائرة

المطلوب تصميم خوارزمية الحل لمسألة حساب محيط ومساحة الدائرة وإعطاء الناتج.

تحليل المسألة:

لحساب محيط ومساحة الدائرة، لا بد من أن يتوفر لدينا طول نصف القطر وبالتالي سيتم تقديم هذه القيمة كمدخل للمسألة المطروحة مع معرفة أن حساب المساحة والمحيط للدائرة يتطلب معرفة قيمة المقدار π وهي قيمة ثابتة تساوي 3.14، وبمعرفة القوانين الرياضية لحساب المقادير المطلوبة يمكن حسابها وتقديمها كمخرجات.

نلخص ماسبق كما يلي:

• القيم المعروفة

- π تساوي 3,14

• المدخلات

- نصف قطر الدائرة.

• الحسابات

- محيط الدائرة = $2 \times \pi \times \text{نصف القطر}$

- مساحة الدائرة = $\pi \times \text{مربع نصف القطر}$

• المخرجات

- محيط الدائرة.

- مساحة الدائرة.

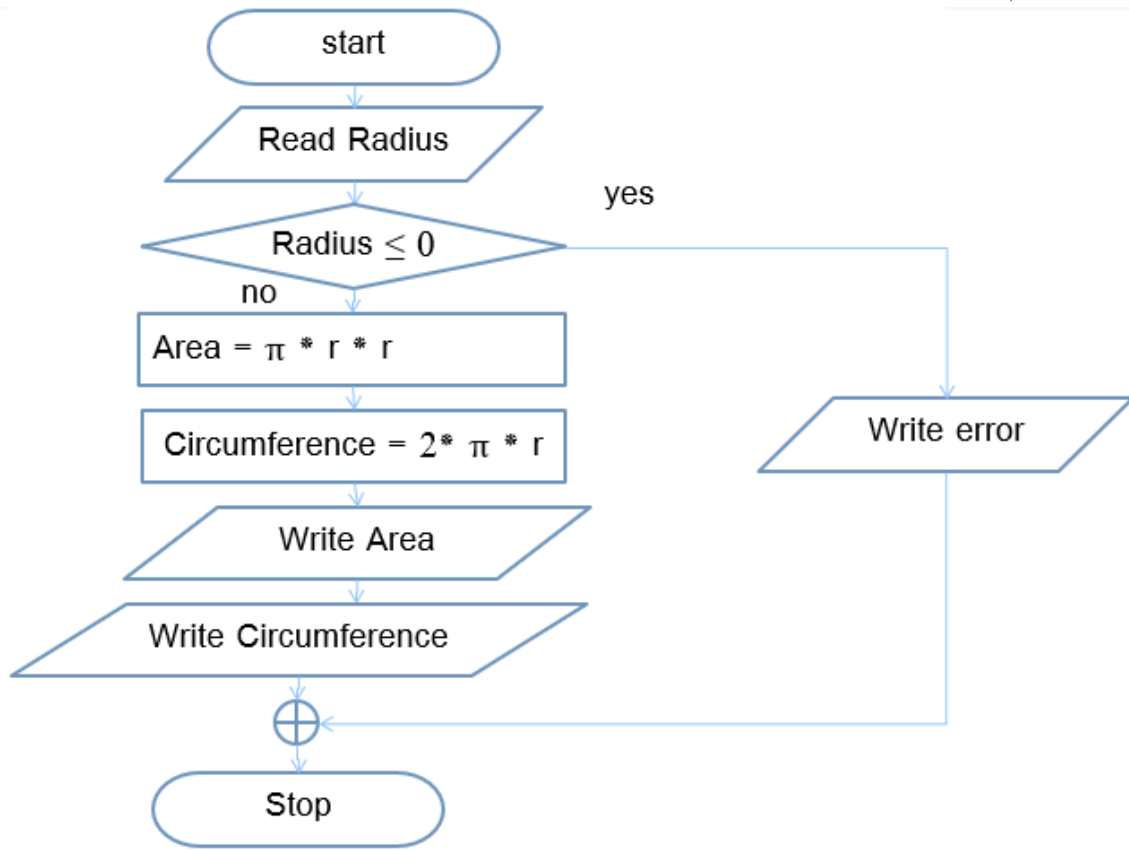
الحل باستخدام شبه الشيفرة:

```
1- START
2- READ Radius
3- IF ( Radius ≤ 0 ) THEN
    1- WRITE "ERROR"
  ELSE
    1- CALCULATE Area =  $\pi * r * r$ 
    2- CALCULATE Circumference =  $2 * \pi * r$ 
    3- WRITE Area
    4- WRITE Circumference
4- STOP
```

ملاحظات:

- ✓ كان من الممكن إضافة أمر خاص بتعيين قيمة π بشكل مسبق (وقد يكون هذا من الممارسات الجيدة).
- ✓ في شبه الشيفرة لسنا معنيين بمعرفة أن المحرف π ليس مدعوماً من قبل لغات البرمجة لأننا لا نكتب بما يتوافق مع لغة برمجة بذاتها، إلا أننا يجب أن ننتبه إلى هذا الأمر أثناء الانتقال إلى الشيفرة البرمجية.

الحل باستخدام المخططات التدفقية:



مابعد الحل:

حاول الاستفادة من هذه الخوارزمية والعودة إلى المراجع الرياضية وإنشاء قائمة بالقوانين والعلاقات التي تعبر عن حساب مساحات وحجوم (وغيرها من المقادير والمحددات) الخاصة بمختلف الأشكال الرياضية (مستطيل، مربع، كرة، أسطوانة، منشور، هرم إلخ) ومن ثم محاولة صياغة الخوارزمية الخاصة بحل مثل هذه المسائل.

5- التمرين الرابع: مسألة العدد الأكبر

المطلوب تصميم خوارزمية الحل لمسألة إيجاد العدد الأكبر بين ثلاثة أعداد وإعطاء الناتج.

تحليل المسألة:

تقوم فكرة الحل على إجراء سلسلة من المقارنات البينية للأعداد الثلاثة التي يجب تقديمها كمدخلات للمسألة وصولاً إلى تحديد العدد الأكبر فيما بينها وتقديمه كمخرج للمسألة.

نلخص ماسبق كما يلي:

• القيم المعروفة

- لا يوجد

• المدخلات

- قيمة العدد الأول.

- قيمة العدد الثاني.

- قيمة العدد الثالث

• الحسابات

- لا يوجد

• المخرجات

- قيمة العدد الأكبر

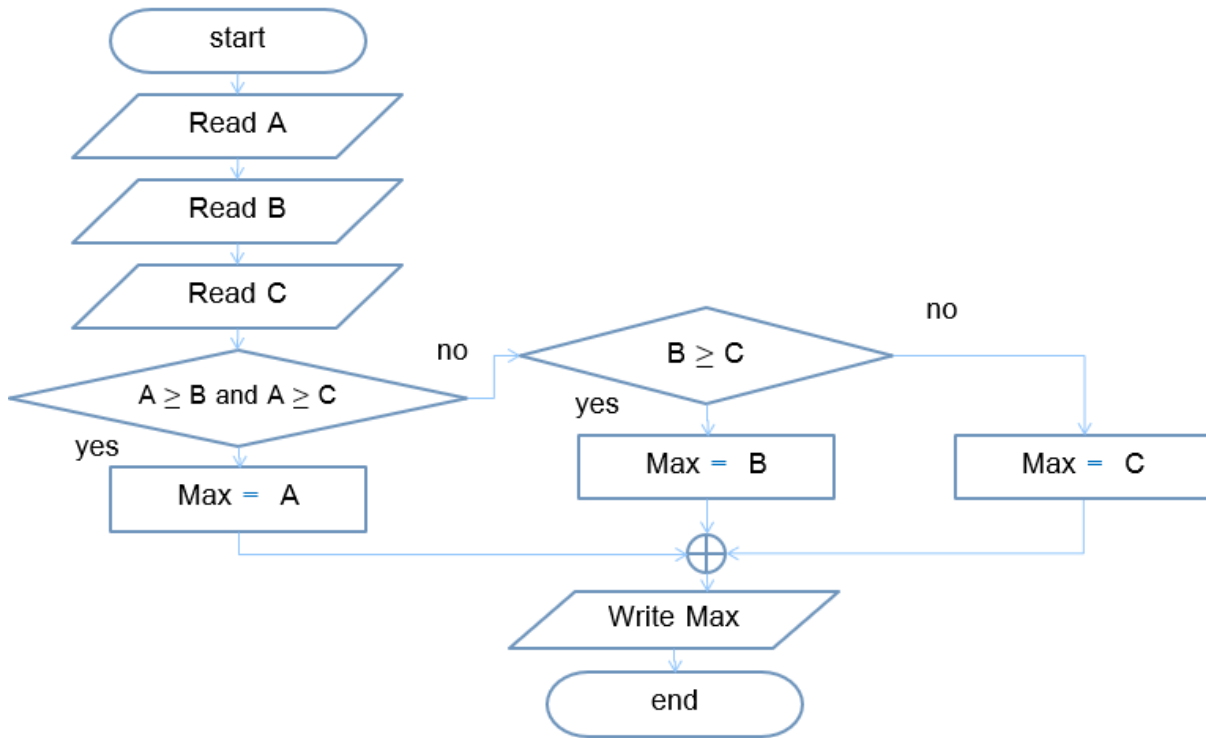
الحل باستخدام شبه الشيفرة:

```
1- START
2- READ A
3- READ B
4- READ C
5- IF (A ≥ B and A ≥ C) THEN
    1- SET Max = A
  ELSE IF (B ≥ C)
    1- SET Max = B
  ELSE
    1- SET Max = C
6- WRITE Max
7- STOP
```

ملاحظات:

- ✓ من شروط الخوارزمية أن تكون خطواتها عنصرية (أبسط ما يمكن) ولهذا السبب قمنا بقراءة القيم الثلاث في ثلاث خطوات منفصلة، رغم شيوع الخطأ المتمثل بدمج الخطوات 2 و 3 و 4 في خطوة واحدة.
- ✓ استخدمنا في الخطوة رقم 5 شرطاً مركباً وكان من الممكن أن نستخدم شرطاً بسيطاً يتضمن مقارنة قيمتين فقط ومتابعة الحل على هذا الأساس.

الحل باستخدام المخططات التدفقية:



ما بعد الحل:

إن الحل السابق يفتقر لصفة العمومية (not Generic) وهذا أمر هام لدى محاولة تصميم الخوارزميات، ويقصد بذلك أن يكون الحل صالحاً لأي مسألة مشابهة مهما اختلف عدد القيم، فمثلاً لا يمكن اعتبار الحل السابق ملائماً لو كان عدد القيم كبيراً (100 قيمة مثلاً) أو غير محدد سلفاً، وفي هذه الحالة الممارسة الأسلم إنشاء بنية تكرارية لإنجاز مثل هذا الحل.

حاول تطوير هذه الخوارزمية لتصبح صالحة لحل المسائل التالية:

- ✓ مسألة إيجاد العدد الأكبر ضمن قائمة من القيم عددها 100 قيمة.
- ✓ مسألة إيجاد العدد الأكبر ضمن مجموعة من الأرقام المدخلة تنتهي بقيمة معينة.

6- التمرين الخامس: مسألة القاسم المشترك الأكبر لعددين

المطلوب تصميم خوارزمية الحل لمسألة إيجاد القاسم المشترك الأكبر لعددين وإعطاء الناتج.

تحليل المسألة:

تقوم فكرة الحل على إجراء سلسلة متكررة من عمليات الطرح للعدد الأصغر من العدد الأكبر بين العددين الذين يجب تقديمهما كمدخلات للمسألة وصولاً إلى تساوي العددين وبالتالي نكون قد وصلنا إلى النتيجة المطلوبة لتقديمها كـ مخرج للمسألة.

مثال: القاسم المشترك الأكبر للعددين 24 و 36 يحسب كما يلي:

a	b
36	24
12	24
12	12

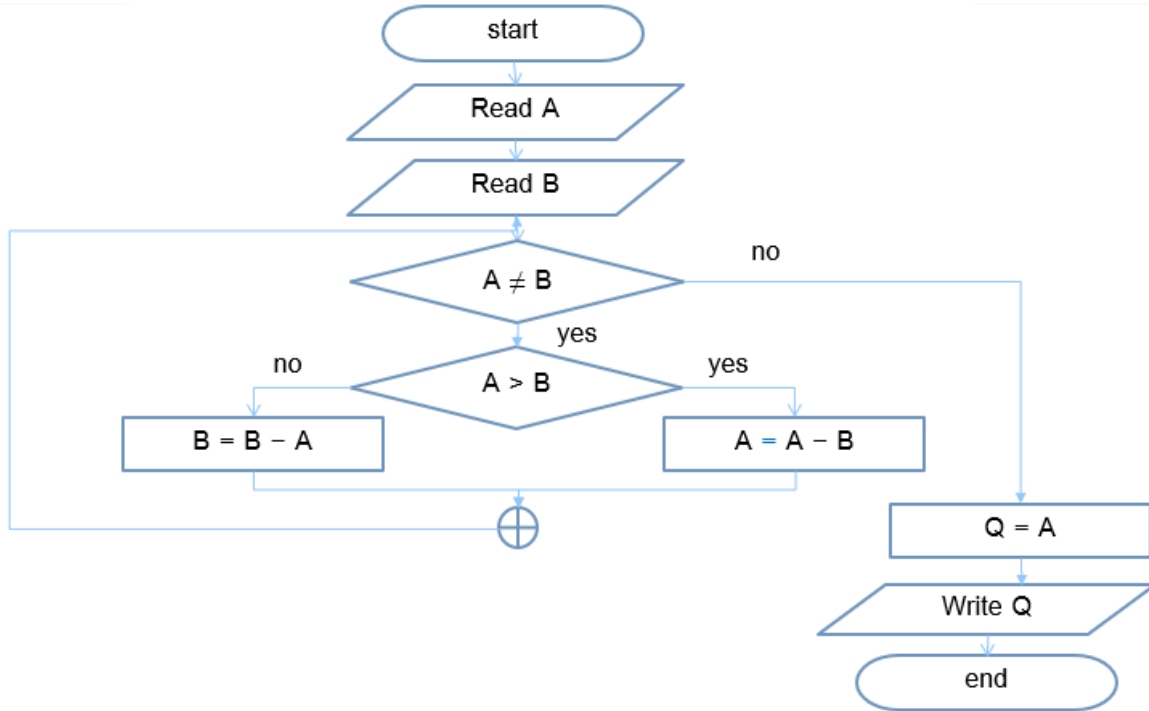
نلخص ماسبق كما يلي:

- القيم المعروفة
 - لا يوجد المدخلات
 - قيمة العدد الأول.
 - قيمة العدد الثاني.
- الحسابات
 - لا يوجد.
- المخرجات
 - قيمة القاسم المشترك

الحل باستخدام شبه الشيفرة:

```
1- START
2- READ A
3- READ B
4- WHILE ( A ≠ B )
5-   1- IF ( A > B ) THEN
       1-1 CALCULATE A = A - B
     ELSE
       1-1 CALCULATE B = B - A
6- SET Q = A
7- WRITE Q
8- STOP
```

الحل باستخدام المخططات التدفقية:



مابعد الحل:

حاول توسيع هذه الخوارزمية أو تعديلها لتصبح صالحة لحل المسائل التالية:

✓ مسألة إيجاد ناتج قسمة عددين باستخدام أسلوب الطرح المتكرر.

✓ مسألة إيجاد ناتج جمع عددين كسريين.

7- التمرين السادس: مسألة جذور معادلة من الدرجة الثانية

المطلوب تصميم خوارزمية الحل لمسألة إيجاد جذور معادلة من الدرجة الثانية وإعطاء الناتج.

تحليل المسألة:

المعادلة من الدرجة الثانية هي معادلة من الشكل:

$$A x^2 + Bx + C = 0$$

تقوم فكرة الحل على حساب مميز المعادلة Δ انطلاقاً من قيم معاملات المعادلة A, B, C التي يجب تقديمهما كمداخلات للمسألة ومن ثم مقارنة هذا المميز مع القيمة الثابتة المعروفة (الصفر) وحساب الجذور بحسب التعابير الحسابية الموافقة لكل حالة.

نلخص ماسبق كما يلي:

- القيم المعروفة
 - القيمة الحدية للمميز Δ هي الصفر
 - المدخلات
 - قيم أمثال المعادلة a, b, c .
 - الحسابات
 - قيمة المميز $\Delta = B^2 - 4 * A * C$
 - قيمة الجذر المضاعف $X = -B / 2 * A$
 - قيمة الجذر الأول $X1 = -B + \sqrt{\Delta} / 2 * A$
 - قيمة الجذر الثاني $X2 = -B - \sqrt{\Delta} / 2 * A$
 - المخرجات
 - قيمة الجذر المضاعف X أو قيم الجذور $X1, X2$.

الحل باستخدام شبه الشيفرة:

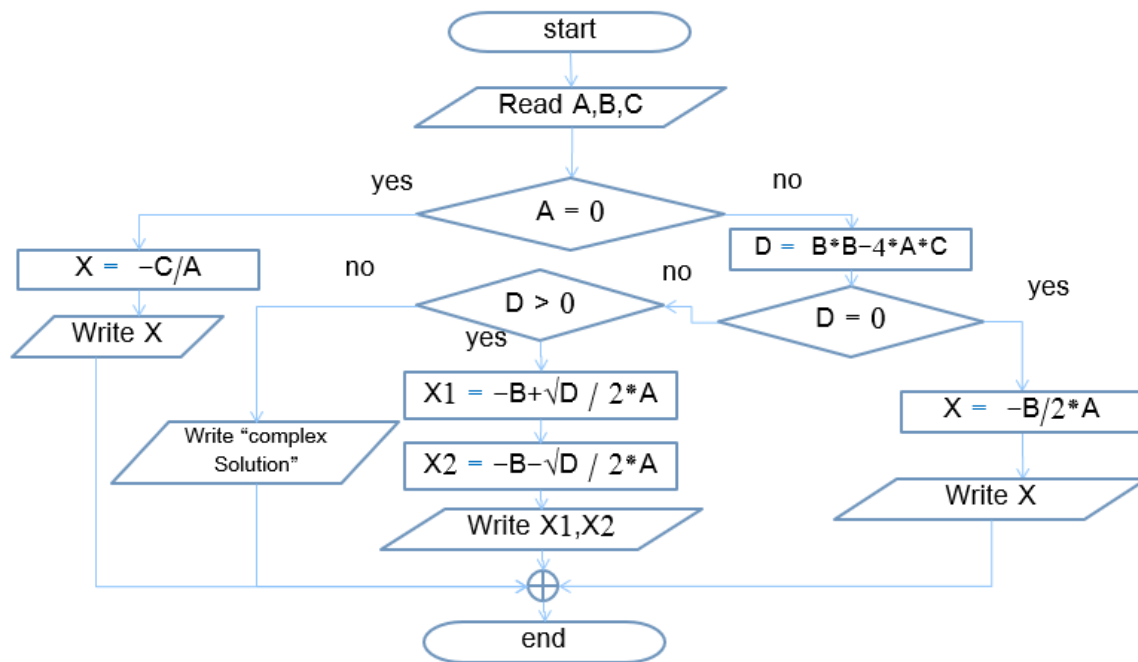
```
1- START
2- READ A,B,C
3- IF (A=0) THEN
  1- CALCULATE X = -C / B
  2- WRITE X
ELSE
  1- CALCULATE D = B*B-4*A*C
  2- IF (D = 0) THEN
    2-1- CALCULATE X = -B / 2*A
    2-2- WRITE X
  ELSE IF (D>0)
    2-1- CALCULATE X1 = -B+√D / 2*A
    2-2- CALCULATE X2 = -B-√D / 2*A
```

2-3- WRITE X1,X2
 ELSE
 2-1- WRITE "complex solution"
 4- STOP

ملاحظات:

✓ تم دمج خطوات الإدخال والإخراج في خطوة واحدة من باب الاختصار الكتابي إلا أن الفصل بينهما هو الممارسة الأسلم.

الحل باستخدام المخططات التدفقية:



مابعد الحل:

حاول التفتيش عن المعالجة الرياضية لحالة كون المميز أصغر من الصفر وبالتالي الحصول على الجذور العقدية للمعادلة، ثم قم بتعديل هذه الخوارزمية بحيث تراعي هذه الحالة.

8- التمرين السابع: مسألة الأعداد المحصورة بين عددين

المطلوب تصميم خوارزمية الحل لمسألة طباعة الأعداد المحصورة بين 1 و 100.

تحليل المسألة:

تقوم فكرة الحل على تكرار الانتقال بين الأعداد المحصورة بين القيمتين الحديتين 1 و 100 وطباعة القيمة التي نقف عندها في كل مرة.

نلخص ماسبق كما يلي:

- القيم المعروفة

- القيمة الحدية الدنيا هي 1.

- القيمة الحدية العليا هي 100.

- المدخلات

- لا يوجد

- الحسابات

- لا يوجد

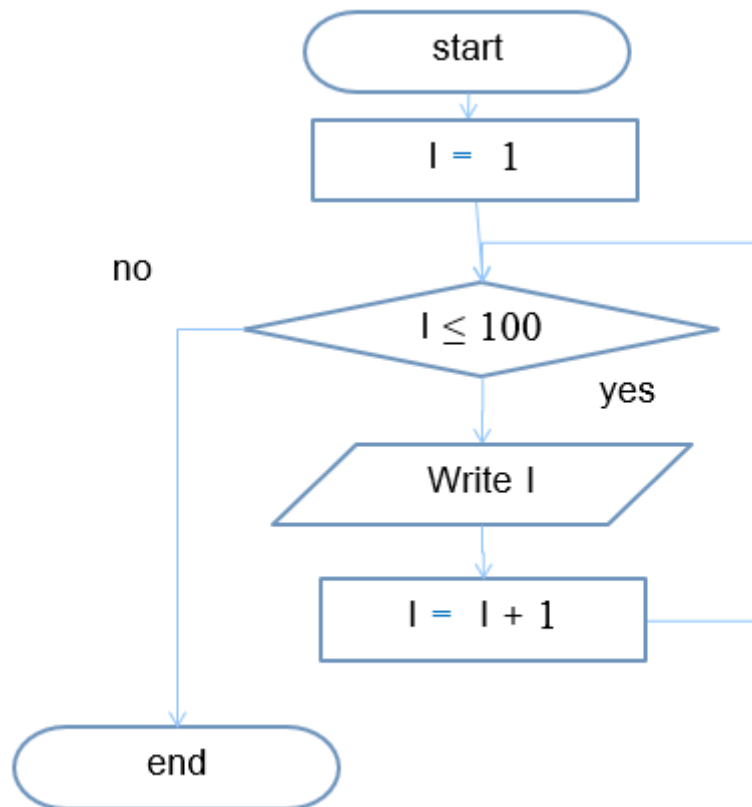
- المخرجات

- الأعداد المحددة.

الحل باستخدام شبه الشيفرة:

```
1- START
2- SET I = 1
3- WHILE I ≤ 100 DO
    1- WRITE I
    2- SET I=I+1
4- STOP
```

الحل باستخدام المخططات التدفقية:



مابعد الحل:

تعتبر مسألة التنقل ضمن مجال معين من القيم تربط بينها علاقة ثابتة (تزايد أو تناقص بمقدار ثابت) وإجراء معالجة مختلفة في كل مرة من أكثر المسائل شبيوعاً في عالم الخوارزميات، وقد طرحت في هذا المثال أبسط أشكالها والذي يتضمن تحديد بداية ونهاية المجال بشكل مسبق.

حاول تعديل هذه الخوارزمية بحيث تكون صالحة لحل المسائل التالية:

- ✓ طباعة الأعداد المحصورة بين قيمتين مدخلتين.
- ✓ طباعة الأعداد المحصورة بين قيمتين مدخلتين بشكل تنازلي.
- ✓ طباعة الأعداد الفردية (أو الزوجية) المحصورة بين قيمتين مدخلتين.
- ✓ طباعة الأعداد التي تقبل القسمة على عدد معين (يحدد أثناء التنفيذ) بين قيمتين مدخلتين.
- ✓ إيجاد المتوسط الحسابي لـ 100 قيمة مدخلة.

9- التمرين الثامن: مسألة حساب العامل

المطلوب تصميم خوارزمية الحل لمسألة حساب العامل لعدد صحيح موجب.

تحليل المسألة:

يحسب عاملي عدد N وفق المعادلة التالية:

$$N! = 1 \times 2 \times 3 \times \dots \times (N-1) \times N$$

تقوم فكرة الحل على تكرار حساب الجداء للأعداد المحصورة بين 1 والعدد N الذي يعتبر كمدخل لهذه المسألة (مع مراعاة كون العدد موجباً) وصولاً لحساب عاملي العدد وتقديمه كـ مخرج للمسألة.

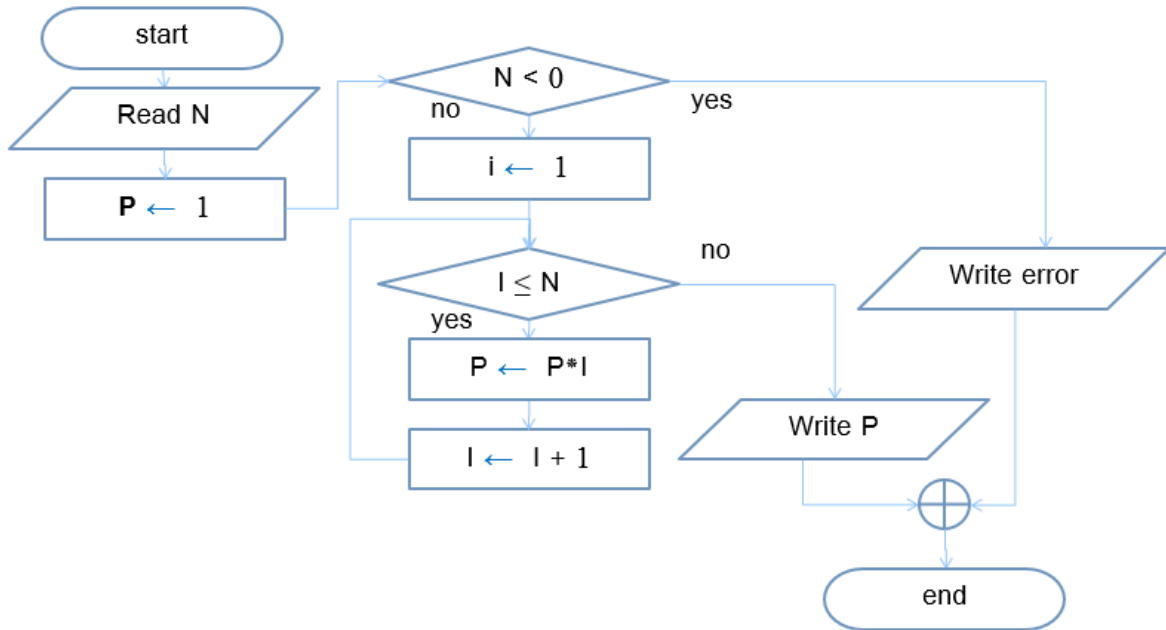
نلخص ماسبق كما يلي:

- القيم المعروفة
 - لا يوجد
- المدخلات
 - العدد N
- الحسابات
 - $N! = 1 * 2 * 3 * \dots * (N-1) * N$
- المخرجات
 - العامل.

الحل باستخدام شبه الشيفرة:

```
1- START
2- READ N
3- SET P = 1
4- IF N < 0 THEN
    1- WRITE "error"
ELSE
    1- SET I = 1
    2- WHILE (I ≤ N) DO
        2-1- CALCULATE P = P*I
        2-2- SET I = I+1
    3- WRITE P
6- STOP
```

الحل باستخدام المخططات التدفقية:



ما بعد الحل:

إن مسألة حساب الجداءات والمجاميع باستخدام أسلوب التكرار هي من المسائل التي تتطلب انتباه خاصاً لحالة تهيئة ناتج العملية بقيمة الحيادي (0 في حال المجاميع ، 1 في حال الجداءات) وهي ذات توظيفات هائلة في حل مسائل أضخم وأكبر.

حاول تعديل هذه الخوارزمية بحيث تكون صالحة لحل المسائل التالية:

- ✓ حساب ناتج رفع عدد x إلى قوة ما n .
- ✓ حساب حدود بعض المتتاليات الشهيرة (متتالية فيبوناتشي، إلخ).