المعلوماتية

الجزء الثالث

م. لبيب الصالح

جامعة حماه | كلية الهندسة المدنية | السنة الأولى

الخوارزميات Algorithms

مفهوم الخوارزمية:

الخوارزمية هي مجموعة من الخطوات المتسلسلة والمحددة والتي تؤدي إلى حل قضية معينة والوصول إلى نتائجها.

سميت الخوارزمية بهذا الاسم نسبة للعالم محمد بن موسى الخوارزمي (780 م – 847 م).

لكي نتعرف أكثر على مفهوم الخوارزمية سوف نناقش قضية **"طرح الطالب سؤالاً ضمن المحاضرة"**:

ونقصد بذلك الخطوات الواجب على الطالب اتباعها لطرح سؤال ضمن المحاضرة حيث يتم اتباع الخطوات المتسلسلة التالية:

1. البداية.
2. الانتظار حتى يصل المحاضر إلى نهاية مقطع كلامي.
3. رفع اليد حتى يؤذَن بالكلام.
4. خفض اليد والمباشرة في طرح السؤال.
5. الاستماع للإجابة، وإذا كانت لا تغطي السؤال يتم إعادة الخطوات من 2 حتى 5 وإلا يتم الانتقال إلى الخطوة 6.
6. النهاية.

تؤدي هذه الخطوات مجتمعة ومرتبة إلى طريقة سليمة لطرح السؤال، لاحظ أنه لا يمكن أن يتم تجاهل خطوة أو تبديل خطوة مكان خطوة أخرى.

تزايدَ الاهتمام بالخوارزميات بشدة مع ظهور الحواسيب لضرورة استخدامها في حل المسائل في جميع المجالات العلمية والتقنية والاجتماعية والاقتصادية والصناعية والتجارية بواسطة الحاسوب.

فلا بد من وضع الخوارزمية اللازمة لحل مسألة معينة قبل كتابة البرنامج الذي يعتمد على هذه الخوارزمية لحل المسألة.

يمكن أن توجد عدة خوارزميات لحل المسألة الواحدة ولكن أفضل هذه الخوارزميات هي التي تصل إلى النتيجة بأقل زمن ممكن (سرعة التنفيذ).

هل يعتبر التعبير الرياضي بحد ذاته خوارزمية؟

لنفترض أن :x عدد ما، ونريد حساب المقدار:

$$y = \frac{3x^2 - 4x + 2}{2x^3 - x^2 - 1}$$

من الواضح أن الحل بسيط جداً، يستطيع إنجازه أي شخص لديه إلمام بسيط بالعمليات الحسابية. فهل يمكن اعتبار "التعبير الرياضي" السابق بحد ذاته خوارزمية لحساب المقدار y ؟

الجواب: **طبعاً لا**

فمع أن التعبير واضح، ويبين العمليات الحسابية اللازمة لحساب المقدار إلاّ أنه لا يعطي تسلسل هذه العمليات.

فيمكن مثلاً أن نبدأ بحساب البسط ثم المقام ومن ثم نقسم البسط على المقام للوصول إلى الجواب، كما يمكن إجراء

العكس بأن نبدأ بحساب المقام ثم البسط وهكذا...

فلكي يصبح التعبير رياضي خوارزمية لا بد أن يقترن بتسلسل تنفيذ عملياته، أي لا بد من إضافة بعض الشروط والقواعد مثل: البدء دوماً وفق أولويات تُحدَّد بحسب نوعية المسألة المطروحة.

تبين الخطوات التالية خوارزمية مقترحة لحل التعبير الرياضي السابق من أجل قيمة ما لـ x :

1- البداية.

2- الحصول على قيمة x.

3- حساب قيمة المقام $B = 2x^3 - x^2 - 1$.

4- هل B تساوي 0 ؟

نعم: اذهب إلى الخطوة 9.

5- حساب قيمة البسط $A = 3x^2 - 4x + 2$.

6- حساب قيمة المقدار $y = A/B$.

7- اطبع y.

8- انتقل إلى الخطوة 10.

9- اطبع " لا يوجد حل (Division By Zero) ".

10-النهاية.

لاحظ أنه في حال كانت قيمة x تعدم المقام فإنه لا حاجة لحساب قيمة البسط (هذه مثلاً أحد القواعد المتبعة في كتابة الخوارزمية السابقة).

توصيف الخوارزمية: Algorithm Specification

يمكن صياغة الخوارزمية بطرق عديدة تتفاوت فيما بينها من حيث دقة التعبير وسهولة الفهم. وأهم الطرق المستخدمة لكتابة الخوارزميات هي:

- صياغتها باللغة الطبيعية، أي اللغة المتداولة كاللغة العربية أو اللغة الإنكليزية وهي الطريقة التلقائية لصياغة الخوارزميات، والتي نستخدمها يومياً تقريباً.
- صياغتها باستخدام لغة رمزية خاصة تسمى pseudo code (الترميز المزيف).
- صياغتها بطريقة بيانِيَّة بواسطة المخطط التدفقي Flowchart Diagram .

الترميز المزيف (pseudo code):

عند تمثيل الخوارزمية باللغة الطبيعية فإن البعض يستخدم الكثير من التفاصيل والبعض الآخر يستخدم القليل منها، فلا يوجد قاعدة معينة لكتابة هذا النوع من الشيفرات. كما أن اللغة الطبيعية تتصف بغنى المعاني وتعتمد في الفهم على خبرة قارئ النص لمعرفة المعنى الدقيق للجملة المكتوبة.

هذه يعني أنه عند قراءة النص من قبل أشخاص متعددين يمكن أن نحصل على معاني مختلفة.

وبما أن نص الخوارزمية يجب ألا يقبل إلا تفسيراً واحداً واضحاً دون لبس أو غموض، لذلك يتم اللجوء إلى لغة رمزية خاصة تسمى الشيفرة المزيفة أو الترميز المزيف pseudo code لكتابة الخوارزمية.

سميت هذه الشيفرة بالشيفرة المزيفة (غير الحقيقية) لأنها ليست مكتوبة وفق لغة برمجة معينة وبالتالي فإنها غير قابلة للتنفيذ على جهاز الكمبيوتر. الخطوة التالية لكتابة الخوارزمية بالكود المزيف هي كتابة البرنامج المعتمد على الخوارزمية وذلك باستخدام كود "حقيقي" وفق لغة برمجة معينة.

قد تؤدي كتابة التعليمات البرمجية (الكود الحقيقي) مباشرة في برامج معقدة نوعاً ما إلى إضاعة الوقت والجهد، والسبب الرئيسي في ذلك هو عدم الاعتماد على خوارزميات صحيحة.

أحد تعريفات برنامج الكمبيوتر:

A computer program generally tries to solve a well-defined problem using a well-defined algorithm.

يمكن القول أن Pseudo code هو وصف "غير رسمي" لبرنامج الكمبيوتر ويجب ترجمته إلى لغة برمجة ما لكي يصبح قابلاً للتنفيذ.

الكلمات المفتاحية Keywords في الترميز المزيف:

في البرمجة: الكلمات المفتاحية هي كلمات ذات دلالة خاصة حيث أن كل لغة برمجة لها كلماتها المفتاحية (الكلمات المحجوزة).

في الترميز المزيف، يتم استخدام بعض الكلمات المفتاحية للإشارة إلى عمليات الإدخال والإخراج وعمليات المعالجة.

بعض الكلمات المفتاحية المستخدمة في الترميز المزيف:

Keyword	Meaning
START	بداية الخوارزمية
END	نهاية الخوارزمية
INPUT	إدخال بيانات من المستخدم
READ / GET	إدخال البيانات من ملف ما
PRINT, DISPLAY, SHOW	إظهار المخرجات على الشاشة أو أداة إخراج أخرى
COMPUTE, CALCULATE, DETERMINE	حساب قيمة تعبير ما
SET, INIT	تهيئة متغير ما بقيمة ابتدائية(Initialization)
INCREMENT	زيادة قيمة متغير ما
DECREMENT	إنقاص قيمة متغير ما
GOTO	الانتقال لتنفيذ خطوة ما (تغيير تسلسل التنفيذ)

العمليات الرياضية Mathematical operations:

العمليات الرياضية هي جزء لا يتجزأ من كتابة الخوارزمية فهي تسمح لنا بمعالجة القيم التي قمنا بتخزينها.

فيما يلي الرموز المعبرة عن العمليات الرياضية الشائعة:

1- إسناد قيمة ما (Assignment):

:=

مثال: $y := x+3$ ، $c := 2\pi r$

2- المقارنة (Comparison):

$=,\ \neq,\ <,\ >,\ <=,\ >=$

3- العمليات الحسابية (Arithmetic):

$+,\ -,\ *,\ /,\ \%$

4- العمليات المنطقية (Logical):

and, or

العبارات الشرطية Conditional statements :

أثناء كتابة الخوارزمية قد نحتاج إلى عبارات تُقيّم (evaluate) التعبيرات (expressions) ومن ثم تنفذ التعليمات بناءً على ما إذا كان التعبير تم تقييمه إلى "صح" أم "خطأ".

أحد أهم الأساليب الشرطية المستخدمة في الترميز المزيف عبارة: **IF — ELSE:**

مثال 1 :

```
START
INPUT x
IF x >=5
        y := 2x-3
        print y
ENDIF
END
```

هنا يتم طلب عدد ما من المستخدم وتخزينه في المتغير X ومن ثم يتم تقييم التعبير (x >=5) فإذا تم تقييمه إلى "صح" يتم تنفيذ التعليمتين: y := 2x-3 و print y

مثال 2 :

```
START
INPUT x
IF x >=5
        y := 2x-3
ELSE
        y := x+1
ENDIF
print y
END
```

هنا يتم طلب عدد ما من المستخدم وتخزينه في المتغير X ومن ثم يتم تقييم التعبير (x >=5) فإذا تم تقييمه إلى "صح" يتم تنفيذ التعليمة: y := 2x-3 وإلا يتم تنفيذ التعليمة y := x+1 . وفي النهاية يتم طباعة قيمة y.

لنعود الآن ونكتب باستخدام الترميز المزيف خوارزمية حل التعبير الرياضي $y = \frac{3x^2-4x+2}{2x^3-x^2-1}$ والتي كنا قد كتبناها سابقاً باللغة الطبيعية:

```
1- START
2- INPUT x
3- B := 2x^3 − x^2 − 1
4- IF B = 0
        GOTO 9
   ENDIF
5- A := 3x^2 − 4x + 2
6- y := A / B
7- PRINT y
8- GOTO 10
9- print "Division by zero"
10-END
```

الآن سنحاول كتابة الخوارزمية السابقة بطريقة ثانية (باستخدام ELSE):

```
1- START
2- INPUT x
3- B := 2x^3 − x^2 − 1
4- IF B = 0
        print "Division by zero"
   ELSE
        A := 3x^2 − 4x + 2
        y := A / B
        PRINT y
   ENDIF
5- END
```

خوارزمية إقليدس Euclid's Algorithm:

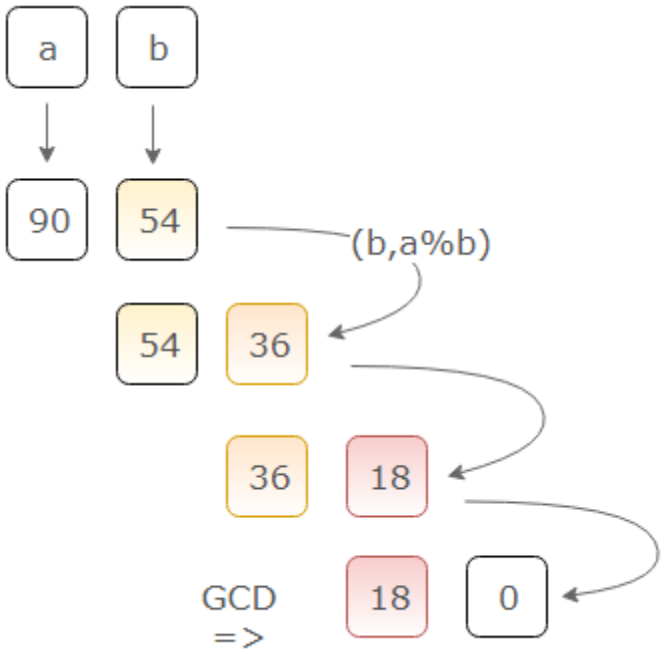
تستخدم خوارزمية إقليدس لإيجاد القاسم المشترك الأكبر Greatest Common Divisor (**GCD**) لعددين (وهو أكبر عدد يقسم كلا العددين دون باقي قسمة - أي الباقي هو 0). تكمن فكرة العالم اليوناني إقليدس في أن القاسم المشترك الأكبر لعددين هو نفسه القاسم المشترك الأكبر للعدد الأصغر منهما وباقي قسمة العدد الأكبر على العدد الأصغر.

وبالتالي فإن خطوات هذه الخوارزمية تبدأ من عددين (**A, B**) مطلوب إيجاد القاسم المشترك الأكبر لهما، **يتم إيجاد باقي قسمة A على B وليكن R. ثم يأخذ B مكان A ويأخذ R مكان B**. وهكذا يتم تكرار هذه العملية حتى نحصل على باقي قسمة يساوي 0، عندها نكون أمام زوج من الأعداد أحدهما صفر وهذا يعني أن العدد الآخر هو القاسم المشترك الأكبر لهما وهو نفسه القاسم المشترك الأكبر للعددين الأصليين.

مثال 1:

نريد استخدام خوارزمية إقليدس لإيجاد القاسم المشترك الأكبر للعددين 90 و54.

$$GCD(A,B) = GCD(90,54) = GCD(54,36) = GCD\ (36, 18) = GCD\ (18, 0) = 18$$



مثال 2:

نريد استخدام خوارزمية إقليدس لإيجاد القاسم المشترك الأكبر للعددين 78 و66.

$$GCD(A,B) = GCD(78,66) = GCD(66,12) = GCD(12,6) = GCD(6,0) = 6$$

الآن وبعد أن أصبحت لدينا فكرة واضحة عن آلية عمل خوارزمية إقليدس سوف نحاول كتابتها بطريقة الترميز المزيف Pseudo code:

خوارزمية إقليدس لإيجاد القاسم المشترك الأكبر لعددين:

تطلب هذه الخوارزمية إدخال عددين ومن ثم تقوم بالعمليات المطلوبة وتطبع القاسم المشترك الأكبر لهما.

```
1- START
2- INPUT A, B
3- IF B = 0
        PRINT A
   ELSE
        R := A%B
        A := B
        B := R
        GOTO 3
   ENDIF
4- END
```

نلاحظ هنا أن **GOTO** أعطتنا إمكانية تكرار تنفيذ خطوة ما (الخطوة 3 في المثال) حتى تَحقق شرط ما (B تساوي 0 في المثال). يمكن أيضاً استخدامها لتكرار عدة خطوات كما سوف نرى لاحقاً.

إضاءة على الخوارزمية السابقة:

في تعليمة IF وضمن قسم ELSE لدينا أربع تعليمات:

```
R := A%B
A := B
B := R
GOTO 3
```

التعليمة الأولى تحسب باقي قسمة العدد A على العدد B وتضع الناتج في R.
التعليمة الثانية تسند قيمة العدد B للعدد A.
التعليمة الثالثة تسند قيمة العدد R للعدد B.
التعليمة الرابعة تنقل التنفيذ إلى الخطوة رقم 3.

هل يمكن أن نبدل مكاني التعليمتين الثانية والثالثة فيما بينهما؟
الجواب بالتأكيد هو لا. لأن ذلك سيؤدي دائماً إلى قيمتين متطابقتين لكل من A و B.

لنعتبر أننا بدلنا مكانيهما ولنأخذ مثالاً نعتبر فيه أن المستخدم أدخل القيم التالية لكل من A و B :

A is 75
B is 20

الخطوة التالية هي: هل B تساوي 0؟ الجواب لا وبالتالي:

R :=A%B	➔	R := 75%20	➔	R is 15
B := R	➔	B:= 15	➔	B is 15
A := B	➔	A := 15	➔	A is 15

لاحظ أننا حصلنا على قيمتين متساويتين لكل من A وB وهي 15(قيمة R). وهذا خاطئ تماماً حيث كان من المفترض أن تكون قيمة A الجديدة هي قيمة B السابقة وهي 20 وأن تكون قيمة B الجديدة هي قيمة R وهي 15.

بينما الترتيب الصحيح للتعليمتين الثانية والثالثة سوف يعطي النتائج الصحيحة التالية:

A is 75

B is 20

"B does **not** equal to 0"

R :=A%B	➔	R := 75%20	➔	R is 15
A := B	➔	A := 20	➔	A is 20
B := R	➔	B:= 15	➔	B is 15

"B does **not** equal to 0"

R :=A%B	➔	R := 20%15	➔	R is 5
A := B	➔	A := 15	➔	A is 15
B := R	➔	B:= 5	➔	B is 5

"B does **not** equal to 0"

R :=A%B	➔	R := 15%5	➔	R is 0
A := B	➔	A := 5	➔	A is 5
B := R	➔	B:= 0	➔	B is 0

B equals to 0 ➔ PRINT A ➔ (It will print "5" which is the GCD)

يتوقف التكرار عندما يتم تقييم التعبير (B = 0) إلى القيمة "صح". وبكلام آخر، يتوقف التكرار عندما يتحقق شرط المساواة (B = 0).
وبالتالي يتم طباعة قيمة A التي صادفت القيمة 0 لـ B. أي سيتم طباعة الرقم "5" على أنه القاسم المشترك الأكبر للعددين المدخلين من المستخدم وهما 75 و20.