

How many shortest-length paths are there to get from your house to the doughnut shop?

↑ up's  
→ right's

$\binom{11}{7} = \binom{11}{4} = 330$  paths

There are six dogs to give 13 tacos. The first and sixth dog get 3 tacos, the second dog gets none, the third dog gets 5 and the fourth dog gets one.

\*\*\*||\*\*\*|\*||\*\*\*|

$A = \{2, 4, 10, 14\}$

$(n) = \frac{n!}{(n-k)!k!}$

$e + 1 = 0$

P	Q	R	P ∨ Q	P ∨ R	(P ∨ Q) ∧ (P ∨ R)
T	T	T	T	T	T
T	T	F	T	T	T
T	F	T	T	T	T
T	F	F	T	F	F
F	T	T	F	T	F
F	T	F	F	F	F
F	F	T	F	T	F
F	F	F	F	F	F

$7, 11, 15, 19, 23, \dots$

$A_1 - A_2 = 4$   
 $A_2 - A_3 = 4$   
 $A_3 - A_4 = 4$   
 $\vdots$   
 $A_{n-1} - A_n = 4$

$3x \forall y (x \leq y + 1 \rightarrow x < 2y)$

$x = c + f = 2$

P.I.E. Example:

$4! = \left[ \binom{6}{1} 5! - \binom{6}{2} 4! + \binom{6}{3} 3! - \binom{6}{4} 2! + \binom{6}{5} 1! \right]$

## أهداف الدرس



يكون الطالب في نهاية هذا الدرس قادراً على:

- 1- تعريف ماهية الكود المزيف
- 2- يدرك أساسيات الكتابة عن طريق الكود المزيف
- 3- يحل مسائل برمجية عن طريق الكود المزيف Pseudocode.

## 1. ما هو الكود المزيف Pseudocode

هي تدوين التعليمات البرمجية بلغة تشبه لغة البرمجة بطريقة مبسطة لتصميم حل لمشكلة برمجية بحيث تكون المصطلحات المستخدمة في الكود المزيف متكوّنة من الكلمات الإنجليزية المستخدمة في حياتنا اليومية. كتابة حلّ مشكلة برمجية عن طريق الكود المزيف بطريقة صحيحة تمكّنك من تحويله إلى أيّ لغة برمجة بكلّ سهولة.

مثال 1

أكتب بلغة Pseudocode برنامجاً يمكّنك من إدخال عدد صحيح وعرض رسالة تبين إن كان العدد المدخل زوجياً أو فردياً.

INPUT number

IF (number % 2 = 0) THEN / يستخدم الرمز % لإيجاد باقي عملية القسمة \*/

msg ← "العدد المدخل عدد زوجي"

ELSE

msg ← "العدد المدخل عدد فردي"

ENDIF

OUTPUT msg

## تعليمات طريقة كتابة الكود المزيف Pseudocode

## ■ المسافة البادئة Indentation

تكتب التعليمات البرمجية الموجودة ضمن جملة برمجية أخرى على مسافة أبعد من هذه الأخيرة.

مثال 2

IF condition THEN

↔ *Instructions*

↔ -----

ENDIF

## ■ التعليقات comments

تحتاج بعض الأحيان إلى كتابة تعليق حول تعليمة برمجية أو معادلة لتتذكر شيئاً معيناً يخص هذه الجملة البرمجية، ولتميز التعليق عن باقي الجمل البرمجية يجب أن يسبق التعليق الرمز: //

مثال 3

discount = price \* (0.25) // الخصم هنا يمثل ربع السعر

## ■ الكلمات المفتاحية Keywords

هي مصطلحات لها معنى في الكود المزيف أو الحلّ الخوارزمي وتكون بأحرف انجليزية كبيرة حتى نُميّزها عن باقي العبارات الأخرى كالمتغيرات.

مثال 4

INPUT / OUTPUT / IF THEN ELSE \ WHILE ENDWHILE \ INTEGER  
\ STRING \ .....

## ■ أنواع البيانات Data Types

اسم النوع	الوصف	أمثلة
INTEGER	عدد صحيح	(3, -4, 20)
FLOAT, REAL	عدد عشري	(3.5, -9.6)
CHAR	حرف	("A", "a", "+", "\$")
STRING	مجموعة من الحروف المتتالية	"HELLO", "Nice To Meet You"
BOOLEAN	قيمة منطقية	(TRUE OR FALSE (1-0))
DATE	تاريخ	01/01/2000

## ■ تعيين المتغيرات والثوابت declaring variables/constants

المقصود بتعيين متغير/ثابت هو إسناد اسم ونوع هذا المتغير.

ملاحظة: بعض لغات البرمجة مثل Python لا تفرض ضرورة تعيين المتغيرات/الثوابت

يفضّل عند كتابة الكود المزيّف Pseudocode تعيين المتغيّرات/الثوابت كالآتي:

**DECLARE VarName:DataType**

مثال 5

**DECLARE FirstName:STRING //**

**DECLARE NumberOfCars:INTEGER**

**DECLARE DateOfBirth:Date**

**CONSTANT discount=0.25**

### ■ العوامل الحسابية mathematics operators

يبيّن الجدول ادناه العوامل الحسابية المستخدمة في كتابة الكود المزيّف Pseudocode

رمز العامل الحسابي	الوصف	مثال
+	للجمع	$x + y$
-	للطرح	$x - y$
*	للضرب	$x * y$
/	للقسمة	$x / y$
DIV	للحصول على الناتج الصحيح من عملية القسمة	$5 \text{ DIV } 2 = 2.5$ the result is <b>2</b>
MOD	للحصول على باقي عملية القسمة	$5 \text{ MOD } 2 = 1$ OR $5 \% 2 = 1$
^	الاس أو القوة	$2^3 = 2 * 2 * 2 = 8$

## ■ عوامل المقارنة comparison operators

رمز عامل المقارنة	الوصف	مثال
=	يساوي	$x = y$
<>	لا يساوي	$x <> y$
<	أصغر من	$x < y$
<=	أصغر من أو يساوي	$x <= y$
>	أكبر من	$x > y$
>=	أكبر من أو يساوي	$x >= y$

## ■ العوامل المنطقية Logic operators

تكون نتيجة الإجراء الذي يتضمن عوامل منطقية هي **true** أو **false**.

رمز العامل المنطقي	الوصف	مثال
and	و	يجب أن يكون طرفي الجملة المنطقية صحيحان لتكون النتيجة <b>true</b> $5 > 2$ and $5 < 10$ : the result is <b>true</b> $3 > 1$ and $3 > 5$ : the result is <b>false</b> $4 < 2$ and $4 <= 4$ : the result is <b>true</b>
or	أو	يكفي أن يكون أحد طرفي الجملة المنطقية صحيحا لتكون نتيجة الجملة المنطقية <b>true</b> $6 > 3$ or $6 > 10$ : the result is <b>true</b> $3 < 1$ or $3 > 5$ : the result is <b>false</b>
not	لا	$\text{not}(3 > 1)$ : the result is <b>false</b> $\text{not}(3 > 1)$ or $3 > 4$ : the result is <b>false</b>

مثال 6

جد نتيجة الجمل المنطقية الآتية، مع العلم أنّ  $x=4$  و  $y=2$  و  $z=6$ :

$x > y$  or  $z > y$  : .....

$\text{not}(x=y)$  and  $z < y$  : .....

$y <> z$  and  $\text{not}(x=y \text{ or } x > (z-y))$  :.....

### الاسناد assignment

هو إجراء يتمثل في اسناد قيمة لمتغير ما.

مثال 7

$x \leftarrow 4$

$y \leftarrow 2.3$

$\text{name} \leftarrow \text{"ahmad"}$

$\text{Total} \leftarrow x + y$

نشاط 2

Variables	Data Types
power	integer
speed	Reel/float/decimal
statement	string



Sale	date
------	------

أسند كل قيمة من القيم الموجودة في الدائرة إلى المتغير المناسب.

---



---



---

### ■ جملة ادخال البيانات INPUT statement

تنفيذ هذه الجملة البرمجية يجعل الحاسوب ينتظر المستخدم لإدخال قيمة معينة من أحد أجهزة الإدخال والتي في معظم الأحيان تكون لوحة المفاتيح.

مثال 8

ينتظر الحاسوب المستخدم لإدخال درجة طالب مثلا // **INPUT degree**

لإدخال الاسم الأول لشخص ما // **INPUT FirstName**

لإدخال إجابة عن سؤال معين // **INPUT Answer**

### ■ جملة عرض المعلومات OUTPUT statement

بعد إدخال البيانات أو المعطيات المطلوبة إلى جهاز الحاسوب، يقوم هذا الأخير بمعالجتها بناء على تعليمات برمجية محددة للحصول على النتائج المرجوة والتي تريد غالبا أن يتم عرضها على شاشة الحاسوب وهو المر الذي نحصل عليه عند استخدامنا جملة العرض أو جملة المخرجات.

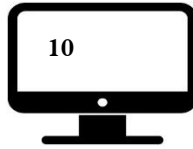
مثال 9



OUTPUT "Welcome"



OUTPUT 5\*2



INPUT Name // المستخدم أحمد يدخل اسمه

OUTPUT Name



أكتب كوداً مزيّفاً Pseudocode يجعل الحاسوب يستقبل قيمتان صحيحتان ويحفظ نتيجة مجموعهما في متغيّر باسم SumVar ويعرض النتيجة على الشاشة.

### ■ الجملة الشرطيّة IF – THEN – ELSE statement

مثال: أرغب في حلّ مسألة بلغة Pseudocode، لبرنامج يمكّنني من إدخال المعدّل الفصلي للطالب وإظهار الرسالة "مبارك عليك النجاح" أو "تستطيع النجاح إذا بذلت مجهوداً أكبر" بناءً على درجة النجاح التي تساوي 60 فما فوق.

الحلّ بلغة Pseudocode:

**INPUT** ("Enter your degree:", degree)

**IF** degree >= 60 **THEN**

**OUTPUT** ("مبارك عليك النجاح")

**ELSE**

**OUTPUT** ("تستطيع النجاح إذا بذلت مجهوداً أكبر.")

**ENDIF**

أكتب بلغة Pseudocode برنامجاً يمكنك من حساب سعر تذكرة الدخول إلى حلبة سباق البحرين للفورمولا 1 بناءً على عمر الشخص بحيث يتمتع الشخص الذي يبلغ من العمر 18 سنة أو أقلّ بتخفيض 25%.

المعطيات:

○ سعر التذكرة 35 دب

▪ الجمل التكرارية / Loop statements

○ جملة WHILE

**WHILE condition DO**

*instructions*

**ENDWHILE**

تمكّني هذه الجملة التكرارية من تكرار التعليمات **instructions**  
عدد غير محدّد من المرات إلى حين تحقّق الشرط **Condition**

مثال: أرغب في حلّ مسألة تتمثّل في إدخال عدد فرديّ وأن يكرّر الحاسوب طلب إدخال هذا العدد إذا كان زوجياً.

الحلّ بلغة Pseudocode:

**INPUT** ("Enter an odd number:",nb)

**WHILE** nb MOD 2 = 0      **DO**

**INPUT** ("Enter an odd number:",nb)

**ENDWHILE**

أكتب بلغة Pseudocode برنامجاً يمكنك من إعادة طلب إدخال عددين صحيحين (قيمتهم الأولى 1 لكل منهما) حتى تصبح نتيجة ضربهما أكبر أو تساوي 1000.

### ○ جملة FOR

**FOR i=start TO end DO**

*instructions*

**ENDFOR**

تمكّني هذه الجملة التكرارية من تكرار التعليمات **instructions** عدد محدّد من المرات حيث يبدأ العدّاد  $i$  من القيمة **start** وحتى القيمة **end** وهي عدد مرات التكرار.

مثال: أرغب في حلّ مسألة تتمثّل في حساب مجموع الأعداد من 15 إلى 20.

الحلّ بلغة Pseudocode:

sum=0

**FOR i=15 to 20 DO**

sum = sum + i

**ENDFOR**

أكتب بلغة Pseudocode برنامجاً يمكنك من حساب وعرض مضاعفات العدد 8 إلى حدود 5 مرات.