

Switch Statement

Switch (Variable)

```

{
  case v1: statements(s1); break;
  case v2: statements(s2); break;
  :
  case vm: statements(sm); break;
  default: // optional
            statements(s);
}

```

تعدم اللفظ في الـ break

عليه ان ياتي في كذا ان
في كذا او لا ياتي

مع استخدام switch في برامج
تعدم اللفظ في الـ break

$$y = \begin{cases} (3x-7) & ; x = -5 \\ (5x^2) & ; x = 2 \text{ or } x = 5 \\ (x-4x) & ; x = -4 \text{ or } x = 4 \end{cases}$$

اكتب برنامج يحسب y

```

#include <iostream.h>
#include <math.h>
int main () {
  int x, y;
  cout << "x=" << cin >> x;
  switch(x) {
  case -5: y = 3*x - 7; break;
  case 2:
  case 5: y = 5*pow(x, 2); break;
  case -4:
  case 4: y = x - 4*x*pow(x, 4); break;
  }
  cout << "y=" << y;
}

```

```
return 0;
}
```

تدريبات: أكتب برنامج يتقارن عددين صحيحين (a,b) بينهما إشارة
 عملية حسابية (5) الخيارات و يطبع نتائج المقارنة حسب العملية
 (-, +, x, /) وفي حال اذغال عملية غير موجودة يطبع الرسالة
 ("insert wrong")

```
#include <iostream.h>
int main ()
float a,b;
char c;
cout << "Enter a, b" ; cin >> a >> b;
switch (c) {
case '+': cout << a << " + " << b << " = " << a+b; break;
case '-': cout << a << " - " << b << " = " << a-b; break;
case 'x': cout << a << " x " << b << " = " << a*x; break;
case '/': if (b != 0) cout << a << " / " << b << " = " << a/b;
else
cout << "Division operation is not possible";
break;
default; cout << "insert wrong";
}
return 0;
}
```

متغيرات الزيادة ++ و النقصان --

```
a++ ; } => a = a + 1 ;
a-- ; } => a = a - 1 ;
```

تستخدم هذه المتغيرات بطريقتين

الزيادة الساكنة أي ++a وتسمى أوت تم عملية الزيادة للمتغير
 ثم استخدامه

```
b=3;
a=++b;
// a=4, b=4
```

مثال

تزداد قيمة b بمقدار واحد ثم تندفع قيمة a الى

الزيادة اوليها من اللاحقة أي $a++$ وتغير أن تم عليه استخدام المتغير ثم اجراء الزيادة عليه

```
b=3;
a=b++;
// a=3, b=4
```

مثال

تندفع قيمة a الى a ثم تزداد قيمة b بمقدار واحد

برنامج: أكتب برنامج يقوم بإسناد القيمة (4) للمتدين a, b, c, d ثم تطبع عليه الزيادة السابقة على a وإسنادها الى c وتطبع عليه الزيادة اللاحقة على b وإسنادها للمتغير d

```
#include <iostream.h>
int main ()
int a, b, c, d;
a=3;
c=++a
cout << "a:" << a << "c:" << c << endl;
b=3;
d=b++;
cout << "b:" << b << "d:" << d << endl;
return 0;
}
```

while (condition) statement; ⇒ مُدینے لے کر،
او

```
while (condition) {
    statement (1);
    statement (2);
    ;
    statement (n);
}
```

برسایغ اکب برسایغ یقوم بارضاد عدد صیغ صوب تمام n و طریض
توا اسم هذا العدد

```
#include <iostream.h>
int main () {
    int n;
    cout << "n: "; cin >> n;
    while (n <= 0)
    {
        cout << "n: "; cin >> n;
    }
    int i = 1;
    while (i <= n)
    {
        if (n % i == 0)
            cout << i << " ";
        i++;
    }
    return 0;
}
```

اکب برسایغ لیرا یقوم من الأعداد الصغیرة الی صغیرة تنسیر بالقیمة الصغیرة
القیمة الصغیرة منقیمة توقف لعلیة الإرضاء و مساوی و طریض
العدد من مساوی لعدد الأعداد

الكتابة الثانية

```
#include <iostream.h>
```

```
int main ()
```

```
{
  int x, n, i;
```

```
i = 0; n = 0;
```

```
cout << "x = " << cin >> x;
```

```
while (x != 0)
```

```
{
  i++;
```

```
n += x;
```

```
cout << "x = " << cin >> x;
```

```
}
if (i != 0)
```

```
{
  avr = (float) n / i;
```

```
cout << "avr = " << avr;
```

```
}
else
```

```
cout << "Set of numbers is empty";
```

```
return 0;
```

```
}
```

بروزت خط
حل التمرين السابق

عدد الأعداد

مجموع الأعداد

(2)

```
do
{
  statement (1);
  statement (2);
```

```
...
statement (m);
```

```
} while (condition)
```

وإذا كان الشرط صحيح يعود

البرنامج لينفذ الشرط

السابق حتى يكون الشرط

غير محقق فيتوقف

تستخدم do while ؟

إذا كان لتنفيذ مرة واحدة

وإن لم يكن الشرط غير صحيح ولا يؤثر

على سير عمل البرنامج

تستخدم مرة واحدة من الأفضل

مثال: عدل البرنامج (١٢) مستخدماً

عملية التكرار do while

لتحقيق عملية الإدخال

الرقمية

منسوخة في هذه الكتابة ملتصقة

for و do while ، انه ملتصقة التكرار

تنفذ الشرط عدد من المرات

while ملتصقة التكرار الأمامي ولكن

do while و for يستخدم

كأنهم ب while ويمكن كتاب

الطوائف do while ، for ب

while ويمكن العكس ليس صحيح

```
#include <iostream.h>
int main ()
int n ;
do
{ cout << "n = " ; cin >> n
}
while (n <= 0) ;
int i = 1 ;
while (i <= n)
{
if (n % i == 0)
cout << i << " " ;
i ++ ;
}
return 0 ;
}
```

عملية التكرار for الحلقة ذات
العداد لها متغير يقوم بعداد
التكرار ، وتكرارها

```
for (initialization ;
condition , increased
statement ;
```

الجزء الثاني الشرط اذ صحت
التوقف لهذا العداد
الجزء الثالث معدل التغير
للعداد

اذا كانت العملية هي عملية واحدة
فورا بعد ، لتوسيع واذا كانت

أتر من عملية نيات نضعهم بين
أقواس
مثل هكذا لآتي

```
for (initialization, condition ;
increase)
{
statement (1) ;
statement (2) ;
...
statement (n) ;
}
```

من استخدام for للعملية التكرارية
اذا كنت تعلم من نبدأ ومن
تنتهي .

البرنامج (١٧) :
أكتب برنامج يقوم بإدخال عدد
صحيح موجب n ، وحساب وطول
n ؟

```
#include <iostream.h>
int main ()
{
int n ; f = 1 ;
do
{ cout << "n = " ; cin >> n ;
} while (n < 0) ;
```

```
for (int i = 0 ; i < n ; i++)
f * = (n - i) ;
```

5

1,1

```
cout << "f=" << f;  
return 0;  
}
```

پ: 18 کتابخانه `<iostream>` مع طیب

در سطح آبر فینه (max) جوده
من الأعداد الصغیرة در ک n

```
#include <iostream.h>  
int main()
```

```
{  
int n, max, x;  
do  
{ cout << "n=" ; cin >> n;  
} while (n <= 0);  
cout << "x=" ; cin >> x;  
max = x;  
for (int i = 2 ; i <= n ; i++)  
{  
cout << "x" << i << " = " ;  
cin >> x;  
if (max < x)  
max = x;  
}  
cout << "max=" << max;  
return 0;  
}
```