

Informatique S1
Programmation C

- *Objectifs* : instruction de contrôle *switch*
- Présentation de l'instruction *switch*
- Usage de l'instruction *break*
- Opérateur ? :

Instruction de contrôle *switch*

- Test multiples valeurs
- Format :

```

switch (expression_int)
{
    case valeur1 : instructions ;
                    break ;
    case valeur2 : instructions ;
                    break ;
    ...
    default : instructions ;
}
    
```

switch (expr_int)

- expression résultant un int
 - o variable de type
 - o int
 - o char (typecast)
 - o long, short, unsigned
 - o expression
- si `expr_int == valeur1`
- `break ;` : termine l'exécution d'un bloc d'instructions
- `default (optionnel) : aucun des cas précédents n'est vrai`

Exemple

```

#include <stdio.h>
int main () {
    char c;
    printf ("Entrez une lettre : ");
    scanf ("%c", &c);
    switch (c) {
        case 'a':
            printf ("voyelle A \n");
            break;
        case 'e':
            printf ("voyelle E \n");
            break;
        case 'i':
            printf ("voyelle I \n");
            break;
        case 'o':
            printf ("voyelle O \n");
            break;
        case 'u':
            printf ("voyelle U \n");
            break;
        default:
            printf ("%c n'est pas une voyelle\n", c );
    }
}
    
```

char c;
switch (c) (Typecast char ← int)

break ;
S'il n'y a pas un break ?!

default ;
S'il n'y a pas un default?!

Exemple

```

1  #include <stdio.h>
2  #include <stdlib.h> //fonction rand
3  #include <time.h>   //fonction time
4
5  /* cm7 : switch (devinette) */
6  int main () {
7      int i,x;
8      srand (time (NULL)); //initialise le generateur
9      x = rand()%11;
10
11     printf ("Entrez i [0-10] : ");
12     scanf ("%d", &i);
13
14     switch (x-i) {
15     case 0: printf ("bingo\n");
16             break;
17     case 1:
18     case -1: printf ("presque\n");
19              break;
20
21     case 2:
22     case -2:
23     case 3:
24     case -3: printf ("proche\n");
25              break;
26     default: printf ("loin\n");
27     }
28     printf ("Reponse : %d\n",x);
29 }
    
```

Instruction **printf** réalisée dans les deux cas :
 $(x-i) == 1$
 $(x-i) == -1$

Opérateur « ? : »

- Opérateur de test « ? : »
 $(test) ? si_vrai : si_faux ;$
- À la place de :

if (a < b)	petit = a;
else	petit = b;
- On peut écrire :

petit = (a < b) ? a : b ;

Exemple

```

1  #include <stdio.h>
2  /* cm 7 : operateur ? : */
3  int main () {
4      int a, b, petit, grand;
5
6      printf ("Entrez a : ");
7      scanf ("%d", &a);
8      printf ("Entrez b : ");
9      scanf ("%d", &b);
10
11     petit = (a<b) ? a : b;
12     grand = (a>b) ? a : b;
13
14     printf ("%d < %d\n", petit, grand);
15 }
    
```

if (a < b)
 petit = a;
 else
 petit = b;

if (a > b)
 grand = a;
 else
 grand = b;

Instructions de contrôle

<h4>Tests</h4> <ul style="list-style-type: none"> if ... else switch ... case ... default 	<h4>Boucles</h4> <ul style="list-style-type: none"> while do ... while for
---	---