

C_facile : Introduction au langage C

A. Le projet cFacile

B. Chapitres du cours

1. Introduction au langage C
2. Algorithmes et langages
3. Structure d'un programme
4. Premiers pas en C
5. Les boucles
 - a. Introduction
 - b. Définition
 - c. Boucles à bornes définies
 - d. Boucles à bornes non définies
 - e. Instructions « for »
 - f. Instruction « while »
 - g. Instruction « do... while »
 - h. Boucles imbriquées
 - i. Choix de la boucle
 - j. Conseils
 - k. Solutions des problèmes en langage C
 - l. Autres Exemples de boucles
 - m. Instruction continue
 - n. **L'instruction break**
 - o. Compléments sur la boucle "for"
6. Tableaux, chaînes, et pointeurs
7. Les fonctions
8. Les structures
9. Allocation dynamique
10. Annexes

C. Exercices

D. Simulations pédagogiques

E. Jeux pédagogiques

F. Liens utiles

L'instruction break

Définition

C'est une instruction de branchement automatique en fin de boucle "do", "while", "for", ou en fin de "switch" (nous avons déjà vu le fonctionnement dans le cas du "switch").

L'instruction break termine l'exécution de l'instruction englobante "do", "while", "for", ou "switch" dans laquelle elle apparaît. L'instruction qui est alors exécutée est celle qui suit immédiatement la fin du "do", "while", "for", ou "switch"

Exemple

```
int i;  
int j;  
for(i=0;i<100;i++)  
{  
    printf("\ni = %d ",i);  
    printf( "\nPour quitter taper -1 : " );  
    scanf("%d", &j);  
    if (c == -1)  
        break;  
}
```

Cette boucle va potentiellement afficher tous les nombres de 0 à 99. Nous demandons à l'utilisateur une lettre.

S'il tape 'Q' nous sortons de la boucle même si elle n'est pas terminée (même si i=100 n'est pas atteint).

Attention

Les bonnes pratiques de la programmation structurée bannissent l'utilisation de « break » au sein d'une boucle « for ».

Cet exemple est donné à titre d'illustration uniquement. Dans ce cas il vaut mieux utiliser une autre boucle en testant deux conditions de sortie « (i<100) » et « (j=-1) ».

Exemple

```
#include <stdio.h>  
#include <math.h>  
#define MAX 100  
for (i=MAX/2; i<MAX;i++)  
{  
    if (((i%2)==0) || ((i%3)==0))||((i%5)==0))  
        continue;
```

```
        else
        if (((i%7)==0) && ((i%11)==0)) break;
        printf("\ni = %d ",i);
    }
    printf("\nsortie : ");
    printf("\n\ti = %d ",i);
```

Ce morceau de code affichera :

i = 53

i = 59

i = 61

i = 67

i = 71

i = 73

La boucle se déroule normalement de MAX/2 à MAX-1 inclus. Elle affiche la valeur de la variable de contrôle de boucle i.

Si i est multiple de 2, 3 ou 5 on ne l'affiche pas car le "continue" passe automatiquement à l'itération suivante en effectuant avant l'instruction 'i++'.

Si i est multiple de 7 et 11 (donc i=77, compte tenu de la boucle), alors l'instruction 'break' fait sortir de la boucle.

