

## ECUE «Introduction à la programmation » - Session 2

14 juin 2017 - Bruno Bouzy  
sans document - durée 1 heure 30

**Exercice 1 (3 points)**

Ecrire un programme `exo1_vote.c` permettant à l'utilisateur d'entrer un nombre de votes « pour » et un nombre de votes « contre » et affichant le vainqueur éventuel avec le pourcentage de succès. La sortie du programme doit respecter la sortie ci-dessous. On suppose que l'utilisateur entre des valeurs strictement positives. On affichera le pourcentage avec un chiffre après la virgule. Les caractères gras sont en entrée du programme. En cas d'égalité le programme affiche EGALITE.

```
nombre de votes POUR ? 200
nombre de votes POUR = 200
nombre de votes CONTRE ? 125
nombre de votes CONTRE = 125
vainqueur : POUR a 61.5
```

**Exercice 2 (4 points)**

Donner la sortie du programme ci-dessous. Pour chaque ligne en **caractères gras**, tenir compte de la **couleur de votre copie (bleu, rouge, vert, jaune)** pour **valoriser a et b** avec les valeurs précisées dans le commentaire de la ligne.

```
#include <stdio.h>
int main() {
    int a = 3; // Couleur de la copie : Bleu:3, Rouge:4, Vert:5, Jaune:7
    int * p = &a;
    int b = *p; printf("1: a = %d, b = %d, *p = %d.\n", a, b, *p);
    a += 4; // Couleur de la copie : Bleu:4, Rouge:5, Vert:7, Jaune:3
    printf("2: a = %d, b = %d, *p = %d.\n", a, b, *p);
    b *= 5; // Couleur de la copie : Bleu:5, Rouge:7, Vert:3, Jaune:4
    printf("3: a = %d, b = %d, *p = %d.\n", a, b, *p);
    int * q = &b; printf("4: a = %d, b = %d, *p = %d, *q = %d.\n", a, b, *p, *q);
    *q *= (*p)++; printf("5: a = %d, b = %d, *p = %d, *q = %d.\n", a, b, *p, *q);
    *q -= ++(*p); printf("6: a = %d, b = %d, *p = %d, *q = %d.\n", a, b, *p, *q);
    p = q; printf("7: a = %d, b = %d, *p = %d, *q = %d.\n", a, b, *p, *q);
    q = &a; printf("8: a = %d, b = %d, *p = %d, *q = %d.\n", a, b, *p, *q);
    return 0;
}
```

**Exercice 3 (3 points)**

Pour chaque question, donner la sortie du programme.

1)

```
void cdt1(int t) { if (t==1) ; printf("A "); printf("B "); }
void cdt2(int t) { if (t==1) printf("A "); printf("B "); }
void cdt3(int t) { if (t==1) { printf("A "); return; } printf("B "); }
int main() {
    printf("0: "); cdt1(0); cdt2(0); cdt3(0);
    printf("\n1: "); cdt1(1); cdt2(1); cdt3(1); printf("\n"); return 0;
}
```

2)

```
void whl1(int t) { int i=t; while (i++<=3); printf("V "); printf("W "); }
void whl2(int t) { int i=t; while (i++<=3) printf("V "); printf("W "); }
void whl3(int t) { int i=t; while (i++<=3) { printf("V "); printf("W "); } }
int main() {
    printf("1: "); whl1(1); printf("\n2: "); whl2(1);
    printf("\n3: "); whl3(1); printf("\n"); return 0;
}
```

3)

```
void for1(int t) { int i=0; for (i=t; i<=3; i++) printf("F "); printf("G "); }
void for2(int t) { int i=0; for ( ; i<=3; i++) printf("F "); printf("G "); }
void for3(int t) { int i=0; for (i=t; ; i++) printf("F "); printf("G "); }
int main() {
    printf("1: "); for1(1); printf("\n2: "); for2(1);
    printf("\n3: "); for3(1); printf("\n"); return 0;
}
```

### Exercice 4 (10 points)

1) Ecrire une fonction `int deIntervalleANombre(int a, int b)` demandant à l'utilisateur un nombre entier appartenant à l'intervalle  $[a, b]$  et retournant ce nombre. Tant que le nombre n'appartient pas à  $[a, b]$ , la fonction re-demande le nombre à l'utilisateur **(2 points)**.

2) Ecrire une fonction `void deTabANombresClavier` prenant en entrée deux entiers  $a$  et  $b$ , un tableau d'entiers `tab` et une longueur `l` de tableau, et remplissant le tableau `tab` avec des nombres distincts appartenant à l'intervalle  $[a, b]$  avec des appels à la fonction `deIntervalleANombre(int a, int b)` ci-dessus. **(2 points)**

3) Ecrire une fonction `void deTabANombresAleatoires` prenant en entrée deux entiers  $a$  et  $b$ , un tableau d'entiers `tab` et une longueur `l` de tableau, et remplissant aléatoirement le tableau `tab` avec des nombres distincts appartenant à l'intervalle  $[a, b]$ , avec des appels appropriés à la fonction `rand()`. **(2 points)**

4) Ecrire un programme `main` simulant le jeu Euromillions respectant l'énoncé suivant. En utilisant un appel à la fonction `deTabANombresClavier`, le programme Euromillions demande à l'utilisateur (le joueur) de remplir un tableau de 5 numéros appartenant à l'intervalle  $[1, 50]$  **(0.5 point)**. Puis le programme affiche les numéros du joueur à l'écran **(0.5 point)**. En utilisant un appel à la fonction `deTabANombresAleatoires`, le programme Euromillions remplit un tableau de 5 numéros aléatoires appartenant à l'intervalle  $[1, 50]$  **(0.5 point)**. Puis le programme affiche les numéros gagnants à l'écran **(0.5 point)**. Enfin, le programme compare les numéros du joueur avec ceux du tirage gagnant : il affiche les numéros corrects et leur nombre **(1 point)**. On veillera à la correction de l'ensemble du programme (constantes, déclarations, variables, tableaux, fonctions, affichages) **(1 point)**. Pour la fonction `deIntervalleANombre`, on respectera les entrée-sorties suivantes:

```
t[0] : x ? (1<=x<=50) 53
x ? (1<=x<=50) 24
t[1] : x ? (1<=x<=50) 9
t[2] : x ? (1<=x<=50) 5
t[2] : x ? (1<=x<=50) 47
t[2] : x ? (1<=x<=50) 38
t[0] = 24, t[1] = 9, t[2] = 5, t[3] = 47, t[4] = 38.
```