

ECUE «Introduction à la programmation »

Contrôle continu n°3 (CC3) 7 janvier 2019
sans document - durée 1 heure 30

Dans tous les exercices, les entrées clavier sont indiquées en caractères gras.

Exercice 1 (4 points)

Ecrire un programme affichant la lettre 'K' de taille T. T est entré au clavier par l'utilisateur. Si $T < 2$, le programme affiche une erreur et retourne -1. Sinon, il affiche le 'K' avec des '*' et des '=', puis retourne T. L'exécution du programme respectera exactement les entrées-sorties ci-dessous. (On remarquera que le K est dessiné avec $2T-1$ lignes de $2T$ caractères).

<p>T ? 4</p> <pre>*=====* *=====*== *==*===== **===== *==*===== *=====*== *=====*</pre>	<p>T ? 3</p> <pre>*=====* *==*== **===== *==*== *=====*</pre>	<p>T ? 2</p> <pre>*==* **== *==*</pre>	<p>T ? 1</p> <pre>erreur: T<2.</pre>
--	--	---	--

Exercice 2 (4 points)

Donner la sortie du programme ci-dessous. Pour chacune des trois lignes avec commentaire, tenir compte de la couleur (Violet, Saumon, Gris ou Jaune clair) de votre copie pour utiliser la valeur précisée dans le commentaire de la ligne à la place de la valeur donnée en gras dans la ligne.

```
#include <stdio.h>
int main() {
    int a = +5; // Violet:-3, Saumon:-3, Gris:+3, Jaune:+5
    int * p = &a;
    int b = *p;
    printf("1: a = %d, b = %d, *p = %d.\n", a, b, *p);
    a *= -3; // Violet:-4, Saumon:+5, Gris:+5, Jaune:-3
    printf("2: a = %d, b = %d, *p = %d.\n", a, b, *p);
    b += -4; // Violet:+7, Saumon:+3, Gris:-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 (12 points)

Une **carte** à jouer est représentée par un nombre entier C entre 0 et 51. Sa **couleur** est le quotient de la division entière de C par 13 (entier entre 0 et 3). Sa **hauteur** est le reste de la division entière de C par 13 (entier entre 0 et 12).

1° Ecrire une procédure affichant une carte. L'affichage se fera sur 2 caractères. 1^{er} caractère : la couleur de la carte ('C' pour 0, 'D' pour 1, 'H' pour 2, 'S' pour 3). 2ème caractère : la hauteur de la carte (h+2 pour h dans {0, 1, 2, 3, ..., 7}, 'T' pour 8, 'J' pour 9, 'Q' pour 10, 'K' pour 11, 'A' pour 12). La signature de la procédure sera void afficheCarte(int carte) ; (Par exemple, si carte vaut 4, la procédure affiche C6 (Six de Trèfle¹). Si carte vaut 38 elle affiche HA (As de Coeur). DK correspond au Roi de Carreau et vaut 24.) **(1 point)**.

2° Un **talon** de 52 cartes est représenté par un tableau de taille 52. Ecrire une procédure initialisant un talon de l cartes avec des cartes ordonnées de 0 à l-1. Sa signature sera void initialiseTalonOrdonne(int * tab, int l) ; **(1 point)**.

Une **main** de joueur contenant entre 0 et 13 cartes est représentée par un tableau de taille 13. Chaque case du tableau correspond à une position de carte dans la main du joueur.

3° Ecrire une procédure affichant un tableau de cartes de taille l. L'affichage se fera sur une seule ligne et sera entre accolades. Sa signature sera void afficheTableau(int * tab, int l) ; Elle appellera afficheCarte autant de fois que nécessaire. **(1 point)**.

4° Etant donné un tableau t1 de taille l en entrée, écrire une procédure remplissant un tableau t2 de taille l mélangé de cartes. On utilisera des appels à rand() % N pour obtenir des nombres entiers aléatoires entre 0 et N-1. Sa signature sera void melange(int * t1, int l, int * t2) ; **(3 points)**.

5° Etant donné un talon t de l cartes et 4 mains de joueurs, écrire une procédure qui distribue les l cartes du talon aux 4 mains. Sa signature sera void distribue(int * t, int l, int * m1, int * m2, int * m3, int * m4) ; **(1 point)**.

6° Etant donné un tableau m de l cartes, écrire une procédure triant m dans le sens croissant. La signature de la procédure sera void tri(int * m, int l) ; **(3 points)**.

7° Ecrire un main() déclarant 6 tableaux : un talon ordonné, un talon mélangé, 4 mains, et appelant les procédures ci-dessus pour : initialiser le talon ordonné, mélanger les cartes dans le talon mélangé, afficher le talon mélangé, distribuer les cartes aux 4 joueurs, trier les 4 mains, les afficher. La sortie du programme aura la forme ci-dessous. **(2 points)**.

Bridge debut:

Talon mélangé:

```
{ S2 SK H3 C7 D7 H5 D4 D2 H7 H6 C4 CT D3 C8 C3 C6 HA C5 D5 S3 S9 C9 SJ HK
CQ DJ CK SQ S6 D8 SA C2 D6 DA DK ST CA D9 HQ S7 S4 S8 DT H2 CJ H9 H8 S5
HJ DQ HT H4 }
```

Mains triées:

```
{ CJ CQ CA D3 D6 D7 H7 HJ HA S2 S4 S6 S9 }
{ C5 C8 C9 D8 D9 DJ DQ DA H5 H6 H9 S8 SK }
{ C3 C4 CK D4 D5 DT DK H3 H8 HT HQ SJ SA }
{ C2 C6 C7 CT D2 H2 H4 HK S3 S5 S7 ST SQ }
```

Bridge fin.

¹C correspond à « Clubs » (Trèfle). D à « Diamonds » (Carreau). H à « Hearts » (Coeur), S à « Spades » (Pique), T à « Ten » (Dix), J à « Jack » (Valet), Q à « Queen » (Dame), K à « King » (Roi), A à « As ».