

Correction de l'examen de Programmation I

Sciences et Technologies du Vivant, Semestre 1

Mercredi 22 décembre 2010

Question 1 – Compréhension de code

Le programme affiche :

A) :-)
B) :-)
C) :-X
D) vdX
E) :-)
F) -))
G) :::
H) :-)
I) XD
:-)
J) 8D8
K):-(
:-)
L):-(
:-)
:-(
:-)
:-(
etc...

Le programme ne s'arrête pas et répète les deux dernières lignes indéfiniment.

Question 2 – Poker

```
(a) void initialiser_jeu(Carte * jeu)
    {
        for(int c = 0; c < 4; c++) {
            for(int n = 2; n <= 14; n++) {
                jeu[c * 13 + n - 2].couleur = c;
                jeu[c * 13 + n - 2].niveau = n;
            }
        }
    }

(b) void melanger_jeu(Carte * jeu)
    {
        const int nb_iterations = 200;

        for(int i = 0; i < nb_iterations; i++) {
            int ind1 = rand() % 52;
            int ind2 = rand() % 52;

            Carte tmp = jeu[ind1];
            jeu[ind1] = jeu[ind2];
```

```

    jeu[ind2] = tmp;
}
}
(c) void distribuer(Carte * jeu,
                  Main * main_joueur1, Main * main_joueur2)
{
    for(int i = 0; i < 5; i++) {
        main_joueur1->cartes[i] = jeu[2 * i];
        main_joueur2->cartes[i] = jeu[2 * i + 1];
    }
}
(d) void affiche_main(Main * m)
{
    for(int i = 0; i < 5; i++)
        affiche_carte(m->cartes + i)
}
(e) bool est_ce_une_flush(Main * m)
{
    return m->cartes[0].couleur == m->cartes[1].couleur &&
           m->cartes[0].couleur == m->cartes[2].couleur &&
           m->cartes[0].couleur == m->cartes[3].couleur &&
           m->cartes[0].couleur == m->cartes[4].couleur;
}
(f) void calcule_compteurs(Main * m)
{
    for(int i = 0; i < 15; i++)
        m->compteurs[i] = 0;

    for(int i = 0; i < 5; i++)
        m->compteurs[ m->cartes[i].niveau ]++;
}
(g) bool est_ce_un_carre(Main * m)
{
    for(int i = 0; i < 15; i++)
        if (m->compteurs[i] == 4)
            return true;

    return false;
}
(h) bool est_ce_une_suite(Main * m)
{
    for(int i = 2; i < 15 - 4; i++)
        if (m->compteurs[i] == 1 &&
            m->compteurs[i + 1] == 1 &&
            m->compteurs[i + 2] == 1 &&
            m->compteurs[i + 3] == 1 &&
            m->compteurs[i + 4] == 1)
            return true;

    return false;
}

```