

Corrigé de l'examen, Programmation I
Sciences et Technologies du Vivant, Semestre 1

Mercredi 17 décembre 2008

Question 1 - Compréhension de code

- (a) A) 1 1
B) 1 2
C) 2
D) 1 2
E) 2 7
F) 5 2 1
G) 1
H) 2
- (b) A) 1 + 3
B) 4
C) 1
D) 4
E) 20
F) 9
- (c) A) n
B) o
C) e
D) +
E) noel

Question 2 - Gestion d'une bibliothèque

- (a)

```
void affiche_livres(Livre *livres) {
    for (int i=0; i<nbr_livres; i++)
        cout << livres[i].titre << " (" << livres[i].auteur << ")" << endl;
}
```
- (b)

```
struct Membre {
    char *nom;
    char *email;
    int age;
    Livre *livre;
};
```

```

(c) void affiche_membres(Membre *membres) {
    for (int i=0; i<nbr_membres; i++) {
        cout << membres[i].nom << " (" << membres[i].email << ")";
        if (membres[i].livre != NULL)
            cout << " - " << membres[i].livre->titre << endl;
        else
            cout << endl;
    }
}

(d) void nouveau_membre(Membre *membre) {
    char buffer[256];
    cout << "Nom: ";
    cin >> buffer;
    membre->nom = new char[strlen(buffer)+1];
    strcpy(membre->nom, buffer);
    cout << "Email: ";
    cin >> buffer;
    membre->email = new char[strlen(buffer)+1];
    strcpy(membre->email, buffer);
    cout << "Age: ";
    cin >> membre->age;
    membre->livre = NULL;
}

(e) void capitalise(char *titre) {
    bool majuscule = true;
    for (unsigned int i=0; i<strlen(titre); i++) {
        if (titre[i] == ' ')
            majuscule = true;
        else {
            if (majuscule)
                if (titre[i] >= 'a' && titre[i] <= 'z')
                    titre[i] += 'A' - 'a';
            majuscule = false;
        }
    }
}

(f) bool emprunte(Membre *m, Livre *l) {
    if (l->emprunteur != NULL) // livre deja emprunte
        return false;

    if (m->livre != NULL) // le membre a deja emprunte un autre livre
        return false;
}

```

```

    m->livre = l;
    l->emprunteur = m;
    Date courant = date_courante();

    courant.mois++;
    if (courant.mois > 12) {
        courant.mois = 1;
        courant.annee ++;
    }
    l->date_limite = courant;

    return true;
}

(g) int difference_dates(Date *d1, Date *d2) {
    int difference = 0;
    difference += 360 * (d1->annee - d2->annee);
    difference += 30 * (d1->mois - d2->mois);
    difference += d1->jour - d2->jour;
    return difference;
}

(h) void verifie_retards(Membre *membres) {
    Date courant = date_courante();
    for (int i=0; i<nbr_membres; i++) {
        if (membres[i].livre != NULL) {
            int difference = difference_dates(&courant,
                                             &(membres[i].livre->date_limite));
            if (difference > 0) // retard
                envoie_mail_retard(membres[i].email,
                                   membres[i].livre->titre,
                                   difference);
        }
    }
}

```