

# Examen Final C++

## Structures de contrôle

---

1. Écrire une fonction qui, étant donné deux entiers  $x$  et  $y$ , calcule leur plus grand commun diviseur (PGCD). Le PGCD est le plus grand entier qui divise exactement  $x$  et  $y$ . (2 points)

## Tableaux

---

2. Écrire une fonction qui prend comme paramètres deux tableaux d'entiers  $A$  et  $B$ , ainsi que leurs tailles respectives. Votre fonction doit décider si le tableau  $A$  est un sous-ensemble de  $B$ , c'est-à-dire, si chaque élément qui apparaît dans  $A$  apparaît aussi dans  $B$ . On suppose que le tableau  $A$  ne contient pas de valeurs répétitives. (2 points)

**Exemple :** Pour les tableaux  $A = [5, 6, 7]$  et  $B = [2, 7, 4, 5, 6]$  votre fonction doit retourner **true**. Pour les tableaux  $A = [5, 6]$  et  $B = [1, 2, 6]$  elle doit retourner **false**.

## Pointeurs et gestion de la mémoire

---

3. Qu'affiche le programme suivant ? (1 point)

```
int b[] = {1,2,0,-2,-1};
int *p[5];
int i;
for(i=0; i<5; i++)
    p[i] = b+b[i];
for(i=0; i<5; i++)
    cout << *(p[i]+i) << endl;
```

4. Considérons le programme suivant:

```
void f()
{
    int *p[5];
    int i,j;
    for(i=0; i<5; i++)
        p[i] = new int[i+1];
    for(i=0; i<5; i++)
        for(j=0;j<i+1;j++)
            p[i][j] = i+j;
    for(i=0; i<5; i++){
        for(j=0;j<i+1;j++)
            cout << p[i][j];
        cout << endl;
    }
    delete [] p;
}
```

Quel va-t-il afficher? Est-ce qu'il y a des erreurs? Si oui, comment les corriger?  
(2 points)

### Listes chaînées

5. Écrire une fonction qui, étant donné un tableau d'entiers, produit une liste chaînée contenant les mêmes éléments du tableau, **dans le même ordre**, et retourne un pointeur sur la tête de cette liste créée. (2 points)

**Note:** Utilisez la définition suivante pour la structure liste chaînée.

```
struct Node { int data; struct Node *next; };
```

6. Écrire une fonction qui, étant donné un pointeur sur la tête d'une liste chaînée, supprime la **seconde moitié** de la liste. Plus précisément, si la liste contient  $s$  éléments, votre fonction doit retenir les  $s/2$  premiers éléments, et supprime le reste. (3 points)

**Exemple :** Pour la liste des éléments 1,2,3,4,5,6 votre programme doit produire la liste 1,2,3, et supprime les éléments 4,5,6. Pour une liste de taille impaire il faut retenir les  $(s+1)/2$  premiers éléments. Par exemple, pour la liste 1,2,3,4,5 il faut produire la liste 1,2,3.

### Classes

7. Écrire une classe **Time**. Chaque objet de type Time stockera un certain nombre d'heures, de minutes et de secondes (tous des entiers).

On vous demande de proposer : (3 points)

1. La définition de la classe Time
2. Un constructeur sans arguments et un constructeur qui prend trois entiers.
3. Une méthode affiche(), qui affiche les champs d'un objet Time.
4. Un opérateur d'addition qui permet d'additionner deux objets Time.

**NB :** Rappelons que les secondes et les minutes ne doivent jamais avoir une valeur >60.

**Exemple :** 75 secondes doit être convertie en 1 minute et 15 secondes.

**NB :** Cette information doit être prise en compte dans le constructeur et dans l'opérateur d'addition.

8. Considérons la classe suivante

```
class A{
int x;
public:
    A() { x=0; cout << "Const" << endl;}
    A(A& a) { x = a.x; cout << "Copy const" << endl; }
```

```
    ~A() { cout << "Dest" << endl; }  
};
```

Qu'affiche le programme suivant? (1 point)

```
void f(A a){ cout << "f" << endl; }  
int main()  
{  
    A a1,a2;  
    f(a1);  
    f(a2);  
}
```

9. Considérons la classe suivante

```
class B{  
    int *x;  
public:  
    B() { x=new int; cout << "Const" << endl;}  
    ~B() { delete x; cout << "Dest" << endl; }  
};
```

Qu'affiche le programme suivant? (1 point)

```
void f(){ B b1,b2; b1=b2; cout << "f" << endl; }  
int main() { f(); f(); f(); }
```

## Héritage

10. Qu'affiche le programme suivant? (2 points)

```
class A {  
protected:  
    int x;  
public:  
    A() { x=0;}  
    A(int x) { this->x = x; }  
    ~A() { cout <<"~A:" << x << endl; }  
    void f() { cout << "A:" << x << endl; }  
};  
  
class B: public A{  
    int y;  
public:  
    B(int x,int y):A(x) { this->y = y; }  
    B() { this->y = 0; }  
    B(int x) {this->y = x; }  
    ~B() { cout << "~B:" << x << ", " << y << endl;}  
    void f() { cout << "B:" << x << ", " << y << endl; }  
};
```

```
int main()
{
    A *p;
    B b1;
    B b2(2);
    B b3(3,4);
    b1.f();
    b2.f();
    b3.f();
    p = &b3;
    p->f();
}
```

**11.** Qu'affiche le programme si on déclare la méthode A::f() comme étant virtual ? **(1 point)**

### **Templates**

---

**12.** Donner une version générique du programme de l'exercice 2 (sous-ensemble). **(1 point)**