

### Rattrapage - 2018/2019 SMI - POOP en C++

#### Exercice 1 :

Donner la sortie du programme suivant, et justifier votre réponse:

```
#include <iostream>
using namespace std;

class T{
    int i;
public:
    T(int n = 0){ cout << "+++ Constructeur \n"; i = n; }
    T( T & v){ cout << "+++ Constructeur par recopie\n"; i = v.i;}
    ~T(){ cout << "--- Destructeur\n";}
};

void f1(T t) {}
void f2(T & t) {}

int main(){
    cout << "DEBUT\n";
    T u;
    T v = u;
    cout << "appel f1 : \n" ;f1(u);
    cout << "appel f2 : \n" ;f2(u);
    T * pt;
    pt = new T(2);
    delete pt;
    cout << "FIN\n";
    return 0;
}
```

#### Exercice 2 :

On considère la classe T suivante :

```
class T{
    int n ;
    int *pn;
public:
    ...
};
```

complétez cette classe pour que le programme suivant :

```
int main(){
    int tab[5] = {1,2,3,4,5};
    T a(tab, 5);
    T b = a;
    b[1] = 0;
    b[3] = 0;
    for(int i = 0; i < a(); i++)
        cout << a[i] << "\t";
    cout << endl;
    for(int i = 0; i < b(); i++)
        cout << b[i] << "\t";
    cout << endl;
    return 0;
}
```

soit exact et produit la sortie suivante :

0	1	2	3	4
1	2	3	4	5
1	0	3	0	5

### Exercice 3 :

On souhaite réaliser une classe Rat qui permet de manipuler les nombres rationnels :

```
class Rat
{
    int num;           // numérateur
    int den;           // dénominateur
public:
    -
};
```

1. Que doit-on ajouter à cette classe pour que les instructions suivantes, produisent les résultats correspondants :

```
Rat a;                // a = 0/1
Rat b(4);              // b = 4/1
Rat c(2,5);            // c = 2/5
Rat d(0,x);            // d = 0/1 (pour tout x entier)
Rat e(x,0);            // e = x/1 (pour tout x entier)
```

2. Définissez les outils nécessaires pour pouvoir écrire :

```
Rat a(5,4);
Rat b;
b = a;
cout << "b = " << b << endl;
```

3. Que doit-on ajouter à la classe Rat pour pouvoir écrire les instructions suivantes :

```
// a1, a2, a3, b et c sont des objets de Rat
a1 = b + c;
a2 = b - c;
a3 = b * c;
```

4. Surchargez les opérateurs de post-incrémentation et pré-incrémentation pour la classe Rat, qui permettent d'ajouter 1 à un rationnel :

```
// a est un objet de Rat
a++;           // a ← a+1
++a;           // a ← a+1
```