

COMPTE RENDU TP C++

William KNITL

Date Rév. 07/02/2024

Page 1 sur 10

COMPTE RENDU GLOBAL TP

*Le code est disponible sur GITHUB à cette adresse :
https://github.com/SHH00/TP_C-*

COMPTE RENDU TP C++

William KNITL

Date Rév. 07/02/2024

Page 2 sur 10

SOMMAIRE

1.	PARTIE 1 : La classe de base – la classe Capteur	3
1.1.	Partie A : Préambule	3
1.2.	Partie B : Introduction.....	3
1.3.	Partie C : Définition des fonctions membres de la classe	3
1.4.	Partie D : Utilisation dans la fonction main	3
1.5.	Partie E : Fonctions d’interface.....	4
1.6.	Partie F : Gestion de la mémoire et tableaux	5
1.7.	Partie G : Nouvelle fonctionnalité	7
2.	PARTIE 2 : Notion d’héritage.....	7
2.1.	Partie H : Préambule	7
2.2.	Partie I : Déclaration de la classe CapteurAnalogique.....	7
2.3.	Partie J : Déclaration et définitions des fonctions annexes	7
2.4.	Partie K : Création de capteurs analogiques	7
2.5.	Partie L : Déclaration et définition de la classe CapteurDigital et de ses membres	8
2.6.	Partie M : Création de capteurs digitaux.....	8
3.	PARTIE 3 : Tableaux et composition.....	8
3.1.	Partie N : Classe Infrastructure	8
3.2.	Partie O : Utilisation	9

COMPTE RENDU TP C++

William KNITL

Date Rév. 07/02/2024

Page 3 sur 10

1. PARTIE 1 : La classe de base – la classe Capteur

1.1. Partie A : Préambule

Cette partie ne nécessite aucune réponse

1.2. Partie B : Introduction

Cette partie ne nécessite aucune réponse

1.3. Partie C : Définition des fonctions membres de la classe

Cette partie ne nécessite aucune réponse, voir le code partie TP1

1.4. Partie D : Utilisation dans la fonction main

Question 6 : Cela permet de suivre étape par étape la création des objets.

Question 7 : Il y a 4 appel de constructeur et 4 de destructeur, c'est normal car Capteur& D = C utilise un alias, donc D est le même objet que C il n'y a donc pas besoin de le construire ou de le détruire. Concernant l'ordre des appels c'est normal on commence par crée le 1er puis le 2nd puis le3eme puis le 4eme et ensuite on détruit chaque couche en commençant par le dernier objet appelé

C:\Users\User\Desktop\tp1\ressources\bin\Debug\TP1.exe

```
Constructeur de capteur de l'objet [0x61fdf0]
Constucteur avec arguments pour la classe capteur de l'objet [0x61fde0]
Constructeur du capteur par copie depuis [0x61fde0] vers le nouvel objet [0x61fdd0]
Constucteur avec arguments pour la classe capteur de l'objet [0xe61750]
Destructeur du capteur [0xe61750]
Destructeur du capteur [0x61fdd0]
Destructeur du capteur [0x61fde0]
Destructeur du capteur [0x61fdf0]

Process returned 0 (0x0)   execution time : 0.121 s
Press any key to continue.
```

Question 8 : Actuellement les données membres de la classe sont statique, il serait intéressant les passés en dynamique avec l'utilisation de new ex: tata = new double il faudrait alors faire un delete tata dans le destructeur.

COMPTE RENDU TP C++

William KNITL

Date Rév. 07/02/2024

Page 4 sur 10

1.5. Partie E : Fonctions d'interface

- La valeur de D est la même que C voire question D-7
- La valeur de B n'est en aucun cas modifiée si l'on modifie la valeur de C ce sont deux objet différents
- Les résultats obtenus sont corrects pour les écarts
- Depuis l'objet E pour appeler la méthode nous utilisons "->" au lieu de "."
- Depuis l'objet D il n'est pas nécessaire de changer la syntaxe normale.
- Si l'on change la syntaxe de la méthode écart en retirant le & cela entraine la copie de l'objet dont on veut faire l'écart avant l'exécution de la méthode, ce qui crée une utilisation non optimisée des ressources.

C:\Users\User\Desktop\tp1\ressources\bin\Debug\TP1.exe

```
Constructeur de capteur de l'objet [0x61fdf0]
Constructeur avec arguments pour la classe capteur de l'objet [0x61fde0]
Constructeur du capteur par copie depuis [0x61fde0] vers le nouvel objet [0x61fdd0]
Constructeur avec arguments pour la classe capteur de l'objet [0x7a6560]
La valeur de A est : -1
La valeur de B est : 10
La valeur de E est : 5
La valeur de D est : 0
La valeur de B est : 10
La valeur de l'ecart A et B est : 11
La valeur de l'ecart C et D est : 0
La valeur de l'ecart E et A est : 1
La valeur de l'ecart E et D est : 0
La valeur de l'ecart D et C est : 0
Destructeur du capteur [0x7a6560]
Destructeur du capteur [0x61fdd0]
Destructeur du capteur [0x61fde0]
Destructeur du capteur [0x61fdf0]

Process returned 0 (0x0)   execution time : 0.020 s
Press any key to continue.
```

COMPTE RENDU TP C++

William KNITL

Date Rév. 07/02/2024

Page 5 sur 10

1.6. Partie F : Gestion de la mémoire et tableaux

Question 1 : L'utilisation de tableaux static entraine l'utilisation du constructeur par copie des 3 objets utilisé précédemment et donc la destruction de ces 3 objets en fin de programme.

Ici aussi cela entraine une mauvaise optimisation des ressources.

```
C:\Users\User\Desktop\tp1\ressources\bin\Debug\TP1.exe
Constructeur de capteur de l'objet [0x61fdf0]
Constructeur de capteur de l'objet [0x61fde0]
Constructeur de capteur de l'objet [0x61fdd0]
Constructeur du capteur par copie depuis [0x61fdf0] vers le nouvel objet [0x408060]
Constructeur du capteur par copie depuis [0x61fde0] vers le nouvel objet [0x408070]
Constructeur du capteur par copie depuis [0x61fdd0] vers le nouvel objet [0x408080]
Destructeur du capteur [0x61fdd0]
Destructeur du capteur [0x61fde0]
Destructeur du capteur [0x61fdf0]
Destructeur du capteur [0x408080]
Destructeur du capteur [0x408070]
Destructeur du capteur [0x408060]

Process returned 0 (0x0)   execution time : 0.050 s
Press any key to continue.
```

Question 2 : L'utilisation de tableau dynamique entraine l'utilisation du constructeur par copie des 3 objets utilisé précédemment et donc la destruction de ces 3 objets en fin de programme.

Ici aussi cela entraine une mauvaise optimisation des ressources.

```
C:\Users\User\Desktop\tp1\ressources\bin\Debug\TP1.exe
Constructeur de capteur de l'objet [0x61fde0]
Constructeur de capteur de l'objet [0x61fdd0]
Constructeur de capteur de l'objet [0x61fdc0]
Constructeur du capteur par copie depuis [0x61fde0] vers le nouvel objet [0x7b6568]
Constructeur du capteur par copie depuis [0x61fdd0] vers le nouvel objet [0x7b6578]
Constructeur du capteur par copie depuis [0x61fdc0] vers le nouvel objet [0x7b6588]
Destructeur du capteur [0x7b6588]
Destructeur du capteur [0x7b6578]
Destructeur du capteur [0x7b6568]
Destructeur du capteur [0x61fdc0]
Destructeur du capteur [0x61fdd0]
Destructeur du capteur [0x61fde0]

Process returned 0 (0x0)   execution time : 0.048 s
Press any key to continue.
```

COMPTE RENDU TP C++

William KNITL

Date Rév. 07/02/2024

Page 6 sur 10

Question 3 & 4:

C:\Users\User\Desktop\tp1\ressources\bin\Debug\TP1.exe

```
Constructeur de capteur de l'objet [0x61fdd0]
Constructeur de capteur de l'objet [0x61fdc0]
Constructeur de capteur de l'objet [0x61fdb0]
Constructeur du capteur par copie depuis [0x61fdd0] vers le nouvel objet [0xa36568]
Constructeur du capteur par copie depuis [0x61fdc0] vers le nouvel objet [0xa36578]
Constructeur du capteur par copie depuis [0x61fdb0] vers le nouvel objet [0xa36588]
Adresse de G[0] : 0xa36568
Adresse de G[1] : 0xa36578
Adresse de G[2] : 0xa36588
Destructeur du capteur [0xa36588]
Destructeur du capteur [0xa36578]
Destructeur du capteur [0xa36568]
Constructeur du capteur par copie depuis [0x61fdd0] vers le nouvel objet [0xa365e8]
Constructeur du capteur par copie depuis [0x61fdc0] vers le nouvel objet [0xa365f8]
Constructeur du capteur par copie depuis [0x61fdb0] vers le nouvel objet [0xa36608]
Adresse de H[0] : 0xa365e8
Destructeur du capteur [0xa36608]
Destructeur du capteur [0xa365f8]
Destructeur du capteur [0xa365e8]
Destructeur du capteur [0x61fdb0]
Destructeur du capteur [0x61fdc0]
Destructeur du capteur [0x61fdd0]

Process returned 0 (0x0)   execution time : 0.050 s
Press any key to continue.
```

Question 5 : Je vois que 6 constructeur et 6 destructeurs sont appelés.

C:\Users\User\Desktop\tp1\ressources\bin\Debug\TP1.exe

```
Constructeur avec arguments pour la classe capteur de l'objet [0x62fdd0]
Constructeur avec arguments pour la classe capteur de l'objet [0x62fde0]
Constructeur avec arguments pour la classe capteur de l'objet [0x62fdf0]
Constructeur du capteur par copie depuis [0x62fdd0] vers le nouvel objet [0x916560]
Constructeur du capteur par copie depuis [0x62fde0] vers le nouvel objet [0x916570]
Constructeur du capteur par copie depuis [0x62fdf0] vers le nouvel objet [0x916580]
Destructeur du capteur [0x62fdf0]
Destructeur du capteur [0x62fde0]
Destructeur du capteur [0x62fdd0]
Destructeur du capteur [0x916560]
Destructeur du capteur [0x916570]
Destructeur du capteur [0x916580]

Process returned 0 (0x0)   execution time : 0.054 s
Press any key to continue.
```

Question 6 : Je choisirais la méthode vector car l'on peut définir les objets au sein du tableau et non en amont, ce qui diminue le nombre de ligne de code.

COMPTE RENDU TP C++

William KNITL

Date Rév. 07/02/2024

Page 7 sur 10

1.7. Partie G : Nouvelle fonctionnalité

Cette partie ne nécessite aucune réponse, voir le code partie TP1

2. PARTIE 2 : Notion d'héritage

2.1. Partie H : Préambule

Cette partie ne nécessite aucune réponse

2.2. Partie I : Déclaration de la classe CapteurAnalogique

Cette partie ne nécessite aucune réponse, voir le code partie TP2

2.3. Partie J : Déclaration et définitions des fonctions annexes

Cette partie ne nécessite aucune réponse, voir le code partie TP2

2.4. Partie K : Création de capteurs analogiques

Oui ça entraîne automatiquement l'appel d'un constructeur de classe, le constructeur par défaut pour les instanciation d'objet par argument mais lors d'une instanciation par copie cela appelle le constructeur par argument de la classe mère. Cela permet une initialisation correcte de l'objet, et permet aussi de respecter la hiérarchie d'héritage.

Les destructeurs fonctionnent de la même façon mais dans l'ordre inverse.

C:\Users\User\Desktop\tp2\bin\Debug\TP2.exe

```
Constructeur de capteur de l'objet [0x63fdb0]
Constructeur avec arguments pour la classe CapteurAnalogique de l'objet [0x63fdb0]
Constructeur de capteur de l'objet [0x63fd50]
Constructeur avec arguments pour la classe CapteurAnalogique de l'objet [0x63fd50]
Constructeur avec arguments pour la classe capteur de l'objet [0x63fcf0]
Constructeur du CapteurAnalogique par copie depuis [0x63fd50] vers le nouvel objet [0x63fcf0]
La valeur analogique mesure est 40 %
La valeur analogique mesure est 32 %
La valeur analogique mesure est 32 %
Destructeur du CapteurAnalogique [0x63fcf0]
Destructeur du capteur [0x63fcf0]
Destructeur du CapteurAnalogique [0x63fd50]
Destructeur du capteur [0x63fd50]
Destructeur du CapteurAnalogique [0x63fdb0]
Destructeur du capteur [0x63fdb0]

Process returned 0 (0x0)   execution time : 0.044 s
Press any key to continue.
```

2.5. Partie L : Déclaration et définition de la classe CapteurDigital et de ses membres

Cette partie ne nécessite aucune réponse, voir le code partie TP2

2.6. Partie M : Création de capteurs digitaux

Un appel du constructeur de classe CapteurDigital entraine automatiquement l'appel d'un constructeur de classe Capteur mais cela n'entraine pas un appel du constructeur de CapteurAnalogique. Les destructeurs fonctionnent de la même façon.

```
C:\Users\User\Desktop\tp2\bin\Debug\TP2.exe
Constructeur de capteur de l'objet [0x63fdc0]
Constucteur avec arguments pour la classe CapteurDigital de l'objet [0x63fdc0]
Constructeur de capteur de l'objet [0x63fd70]
Constucteur avec arguments pour la classe CapteurDigital de l'objet [0x63fd70]
Constucteur avec arguments pour la classe capteur de l'objet [0x63fd20]
Constructeur du CapteurDigital par copie depuis [0x63fd70] vers le nouvel objet [0x63fd20]
La valeur digital mesure est 78.125 %
La valeur digital mesure est 19.5312 %
La valeur digital mesure est 19.5312 %
Destructeur du CapteurDigital [0x63fd20]
Destructeur du capteur [0x63fd20]
Destructeur du CapteurDigital [0x63fd70]
Destructeur du capteur [0x63fd70]
Destructeur du CapteurDigital [0x63fdc0]
Destructeur du capteur [0x63fdc0]

Process returned 0 (0x0)   execution time : 0.049 s
Press any key to continue.
```

3. PARTIE 3 : Tableaux et composition

3.1. Partie N : Classe Infrastructure

Cette partie ne nécessite aucune réponse, voir le code partie TP3

COMPTE RENDU TP C++

William KNITL

Date Rév. 07/02/2024

Page 9 sur 10

3.2. Partie O : Utilisation

Question 2 & 3 : Les valeurs sont donc valides

```
La valeur du Capteur d'humidite est : -1
La valeur du Capteur d'humidite est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur d'humidite est : -1
La valeur du Capteur d'humidite est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur d'humidite est : -1
La valeur du Capteur d'humidite est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur moyenne de l'humidite est : -1
La valeur moyenne de la temperature est : -1
La valeur moyenne de l'humidite est : -1
La valeur moyenne de la temperature est : -1
La valeur moyenne de l'humidite est : -1
La valeur moyenne de la temperature est : -1
```

Question 4 : Les valeurs sont les mêmes que précédemment.

COMPTE RENDU TP C++

William KNITL

Date Rév. 07/02/2024

Page 10 sur 10

```
La valeur du Capteur d'humidite est : -1
La valeur du Capteur d'humidite est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur d'humidite est : -1
La valeur du Capteur d'humidite est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur d'humidite est : -1
La valeur du Capteur d'humidite est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur moyenne de l'humidite est : -1
La valeur moyenne de la temperature est : -1
La valeur moyenne de l'humidite est : -1
La valeur moyenne de la temperature est : -1
La valeur moyenne de l'humidite est : -1
La valeur moyenne de la temperature est : -1
Constucteur avec arguments pour la classe capteur de l'objet [0xf46520]
Constructeur du CapteurAnalogique par copie depuis [0xf46300] vers le nouvel objet [0xf46520]
Constucteur avec arguments pour la classe capteur de l'objet [0xf46580]
Constructeur du CapteurAnalogique par copie depuis [0xf46360] vers le nouvel objet [0xf46580]
Constucteur avec arguments pour la classe capteur de l'objet [0xf46800]
Constructeur du CapteurDigital par copie depuis [0xf463d0] vers le nouvel objet [0xf46800]
Constucteur avec arguments pour la classe capteur de l'objet [0xf46850]
Constructeur du CapteurDigital par copie depuis [0xf46420] vers le nouvel objet [0xf46850]
Constucteur avec arguments pour la classe capteur de l'objet [0xf468a0]
Constructeur du CapteurDigital par copie depuis [0xf46470] vers le nouvel objet [0xf468a0]
Constucteur avec arguments pour la classe capteur de l'objet [0xf468f0]
Constructeur du CapteurDigital par copie depuis [0xf464c0] vers le nouvel objet [0xf468f0]
Constructeur par copie de Infrastructure de l'objet [0x67fa40]
La valeur du Capteur d'humidite est : -1
La valeur du Capteur d'humidite est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur du Capteur de temperature est : -1
La valeur moyenne de l'humidite est : -1
La valeur moyenne de la temperature est : -1
```