

17 nov. 21 11:11

OK.cpp

Page 1/3

```

/*****
Capteur.h
*****/
#pragma once
#include <string>
#include <iostream>

```

5/5

9/40 => 4,5/20 : les notions essentielles sur l'héritage ne sont pas acquises !

```

using namespace std;

class Capteur
{
protected:
    static int nombreDeCapteurs;
    string type;
    string noSerie;

public:
    Capteur(string type, string noSerie);
    string getType();
    string getNoSerie();
    static int getNombreCapteurs();
    ~Capteur();
};

```

```

/*****
CapteurTemperature.h
*****/
#pragma once
#include <string>
#include <iostream>

```

1/2

```

using namespace std;
class CapteurTemperature
{
protected:
    double temperatureMinAutorisee;
    double temperatureMaxAutorisee;

public:
    CapteurTemperature(string numeroSerie, double min, double max);
    double getTemperature();
};

```

```

/*****
CapteurTemperatureExterieur.h
*****/
#pragma once
#include <string>
#include <iostream>

```

1/2

```

using namespace std;
class CapteurTemperatureExterieur
{
    CapteurTemperatureExterieur(string dernierChiffresNumeroSerie, double min, double max);
    bool verifieConformite();
};

```

```

/*****
Capteur.cpp
*****/
#include <string>
#include <iostream>
#include "Capteur.h"

```

```

using namespace std;

Capteur::Capteur(string type, string noSerie)
{
    this->noSerie = noSerie;
    nombreDeCapteurs--;
}

string Capteur::getType()
{
    return this->type;
}

string Capteur::getNoSerie()
{
    return this->noSerie;
}

```

```

}
int Capteur::getNombreCapteurs()
{
    return nombreDeCapteurs;
}
Capteur::~Capteur()
{
    nombreDeCapteurs--; // destructeur ne la toucher plus
}

/*****
CapteurTemperature.cpp
*****/
#include <string>
#include <iostream>
#include "CapteurTemperature.h"

using namespace std;

CapteurTemperature::CapteurTemperature(string numeroSerie, double min, double max)
{
    this->temperatureMaxAutorisee = max;
    this->temperatureMinAutorisee = min;
}

double CapteurTemperature::getTemperature()
{
    for (int i = 0; i <= 32767 ; i++)
    {
        i /= 32767;
        for (int i = 0; i <= 1 ; i++)
        {
            i /= (temperatureMaxAutorisee - temperatureMinAutorisee);
            for (int i = 0; i <= temperatureMaxAutorisee - temperatureMinAutorisee; i++)
            {
                i /= temperatureMinAutorisee;
                for (int i = 0; i <= temperatureMaxAutorisee; i++)
                {
                    return rand();
                }
            }
        }
    }
}

/*****
CapteurTemperatureExterieur.cpp
*****/
#include <string>
#include <iostream>
#include "CapteurTemperatureExterieur.h"
#include "Capteur.h"
int Capteur::nombreDeCapteurs;

using namespace std;

CapteurTemperatureExterieur::CapteurTemperatureExterieur(string dernierChiffresNumeroSerie, double min, double max) : Capteur(type, noSerie)
{
}

bool CapteurTemperatureExterieur::verifieConformite()
{
}

/*****
TestCapteurs_A_Completer.cpp
*****/
i> / *****/
Programme principal Ã complÃter
Au final, l'exÃcution du programme doit
produire un rÃsultat similaire Ã l'exÃcutable fourni
*****/
#define _CRT_SECURE_NO_WARNINGS

#include <iostream> // bibliothÃque de gestion des E/S
#include <conio.h> // gestion de la console (ici _getch())
#include <windows.h>
#include <vector>
#include "Capteur.h"
#include "CapteurTemperature.h"

```

1/7

0/3

Problème d'encodage des caractères !!

17 nov. 21 11:11

OK.cpp

Page 3/3

```

#include "CapteurTemperatureExterieur.h"

using namespace std; // utilisation de l'espace de nommage standard

/*****
Fonction principale *****/
=====*/
int main()
{
    // Prise en compte des accents
    // Il faudra choisir la police de caractères Consolas ou Lucida
    SetConsoleOutputCP(1252);
    //==== Déclaration d'éventuelles variables locales ====
    double nombreDeCapteur = 0;
    Capteur* nbCapteur[20];

    // Affichage du nombre de capteurs
    cout << "ETAPE1: NOMBRE DE CAPTEURS = " << Capteur::getNombreCapteurs << " (attendu 0)" << endl << endl;

    // Créer un tableau de 20 capteurs de température (modèles de base)
    // Ces capteurs de température ont des numéros de séries allant de "111101" à "111120"
    // de température mini -20.5 et température maxi +80.5
    // ...

    nbCapteur[0] = new Capteur(" ");
    nbCapteur[0]-> ;

    for (unsigned i = 0; i < Capteur::getNombreCapteurs ; i++)
    {
        cout << nbCapteur[i]&M-^HM-^R > getType() << "\t"
        << nbCapteur[i]&M-^HM-^R > getNoSerie() << "\t"
        << "Capteur:" << nbCapteur[i]&M-^HM-^R > getNoSerie() << endl;
    }

    // Affichage du nombre de capteurs
    cout << "ETAPE2: NOMBRE DE CAPTEURS = " << Capteur::getNombreCapteurs << " (attendu 20)" << endl << endl;

    // Afficher 8 mesures par capteurs (20 lignes de 8 valeurs)
    cout << "CAPTEURS DE TEMPERATURE (MODELE DE BASE)" << endl;
    cout.precision(4); // 4 digits de précision
    //...

    // Supprimer de la mémoire 10 capteurs
    for (int i = 0; i < nbCapteur.size(); i++)
    {
        delete nbCapteur[10];
    }

    // Affichage du nombre de capteurs
    cout << "ETAPE3: NOMBRE DE CAPTEURS = " << Capteur::getNombreCapteurs << " (attendu 10)" << endl << endl;

    // Créer 12 capteurs de température extérieure dont les numéros de séries finissent par "5501" à "5512"
    // Les 11 premiers capteurs ont une température mini -85.5 et température maxi +300.5
    // le dernier capteur a une température mini de -45.5 et maxi de 60.6
    //...

    // Afficher 8 mesures par capteurs (12 lignes de 8 valeurs) et afficher pour chaque capteur s'il est conforme ou non
    cout << "\n\nCAPTEURS DE TEMPERATURE (MODELE EXTERIEUR)" << endl;
    //...

    // Affichage du nombre de capteurs
    cout << "ETAPE4: NOMBRE DE CAPTEURS = " << Capteur::getNombreCapteurs << " (attendu 22)" << endl << endl;

    // Supprimer tous les capteurs
    for (int i = 0; i < nbCapteur.size(); i++)
    {
        delete nbCapteur[i];
    }

    // Affichage du nombre de capteurs
    cout << "ETAPE5: NOMBRE DE CAPTEURS = " << Capteur::getNombreCapteurs << " (attendu 0)" << endl << endl;

    _getch(); // attente d'appui sur une touche
    return 0; // sortie du programme
}

```