Table des matières :

[1 Programme C++ 2](#_Toc94627984)

[1.1 LectureTrame.h 2](#_Toc94627985)

[1.2 LectureTrame.cpp 3](#_Toc94627986)

[1.3 Main.cpp 4](#_Toc94627987)

# Programme C++

Le programme réalisé en C++ sert à fragmenter les trames reçues par l’anémomètre pour y trouver la direction, la vitesse et la température. Une fois les trames segmenter et ces valeurs trouver, on les fait afficher.

## LectureTrame.h

#include <string>

**using** **namespace** std;

**class** CV7OEMFR

Vit, Temp et Dir sont les trames dans lesquels nous extraire la Vitesse, la Température et la Direction.

{

**private**:

string Vit, Temp, Dir, Direction;

**float** Vitesse, Temperature;

**void** splitString(string trame, string delim = ",");

**public**:

CV7OEMFR(string trameVit = "", string trameTemp = "", string trameDir = "");

Ces trois fonctions servent à retourner les valeurs demandées.

**float** getVitesse();

**float** getTemperature();

string getDirection();

};

## LectureTrame.cpp

#include "CV7OEMFR.h"

CV7OEMFR::CV7OEMFR(string trameVit, string trameTemp, string trameDir)

{

Dans le constructeur, nous trions les trames reçus par variables adaptée

**this**->Vit = trameVit;

**this**->Temp = trameTemp;

**this**->Dir = trameDir;

}

**float** CV7OEMFR::getVitesse()

{

splitString(Vit);

**return** **this**->Vitesse;

Les fonctions « get » permet de renvoyer les valeurs demandées. Grâce à la fonction splitString.

}

**float** CV7OEMFR::getTemperature()

{

splitString(Temp);

**return** **this**->Temperature;

}

string CV7OEMFR::getDirection()

{

splitString(Dir);

**return** **this**->Direction;

Fonction permettant de fragmenter les trames. Celle-ci sont fragmenter dès qu’une « , » est rencontrée

}

**void** CV7OEMFR::splitString(string trame, string delim)

{

**int** start = 0;

**int** end = trame.find(delim);

**int** flag = 0;

**while** (end != -1)

{

start = end + delim.size();

end = trame.find(delim, start);

flag++ ;

**if** (trame == **this**->Vit)

{

**if** (flag == 3)

{

Vitesse = stof(trame.substr(start, end - start));

}

}

**if** (trame == **this**->Dir)

{

**if** (flag == 2)

{

Direction = (trame.substr(start, end - start));

}

}

**if** (trame == **this**->Temp)

{

**if** (flag == 2)

{

Temperature = stof(trame.substr(start, end - start));

}

}

}

}

## Main.cpp

#include "CV7OEMFR.h"

#include <iostream>

**using** **namespace** std;

**int** **main**()

{

Trames déclarées

string Direction = "$IIMWV,344.0,R,005.9,M,A\*31<CR><LF>";

string Vitesse = "$IIMWV,344.0,R,005.9,M,A\*31<CR><LF>";

string Temperature = "WIXDR,C,026.0,C,,\*56<CR><LF>";

CV7OEMFR\* trame;

trame = **new** CV7OEMFR(Vitesse,Temperature,Direction);

Affichage Direction, vitesse et température.

cout << "Trame Direction : " << trame->getDirection();

cout << endl << "\nTrame Vitesse : " << trame->getVitesse();

cout << endl << "\nTrame Temperature : " << trame->getTemperature() << endl;

**return** 0;

}