Les questions à se poser avant de commencer à coder :

La première question à se poser est de savoir si on va faire de la programmation fonctionnelle ou orienté objet.

# Programmation fonctionnelle

Si on fait de la **programmation fonctionnelle**, on n’a pas besoin de définir des classes, généralement, on se contentera donc **d’un seul fichier source (.cpp)** dans lequel on va écrire nos fonctions.

La fonction principale qui sera exécuté est la fonction **main**(), donc c’est dans cette fonction que l’on va tester les fonctions que l’on aura écrit.

Exemple :

Voir les exercices précédents sur les tableaux, vector, …

<https://github.com/AurelienDidier/Exercice_Cpp>

//Exercice 3.1

int areDivisible(int x, int y){

return x%y ==0 || y%x==0;

}

//Exercice 3.2

void swap(int &x, int &y){

int c=x;

x=y;

y=c;

}

int main() {

//Exercice 1.1

bool pair = isPair(6);

# Programmation orienté objet

Si on a besoin de faire de la programmation orienté objet, alors il faut commencer par penser à la structure du code (sous forme de diagramme UML ou sur feuille) que l’on va utiliser.

En ensuite, on créer les headers (fichiers .h) pour décrire nos classes, leurs attributs et leurs méthodes.

|  |
| --- |
|  |

class Arbre {

public:

Etat etat;

void brule();

virtual void propage(Foret &foret, int ligne, int colonne);

virtual string getImage();

};

Une fois que cela est fait, on créer également les fichiers .cpp correspondant pour décrire le comportement des méthodes décrites dans le .h.

void Arbre::brule(){

if (etat== Etat::enVie){

etat= Etat::enFeu;

}

}

void Arbre::propage(Foret &foret, int ligne, int colonne){

}

string Arbre::getImage(){

if (etat==Etat::enVie){

return " A ";

}

if (etat==Etat::enFeu){

return " \* ";

}

if (etat==Etat::enCendre){

return " # ";

}

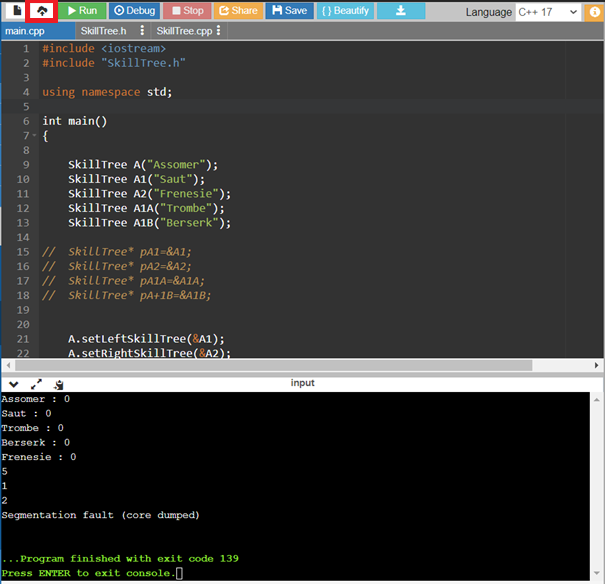
return " ";

}

# Installation

### Sur Online GDB :

Utiliser le upload file pour récupérer les fichiers que vous voulez importer.



Une fois les trois fichiers importer, faire run. Pour exécuter la méthode main et vérifier que tout se passe bien.

### Avec Eclipse :

1. Télécharger Eclipse :

<https://www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/2021-03/R/eclipse-cpp-2021-03-R-win32-x86_64.zip>

Puis dézippez le chez vous.

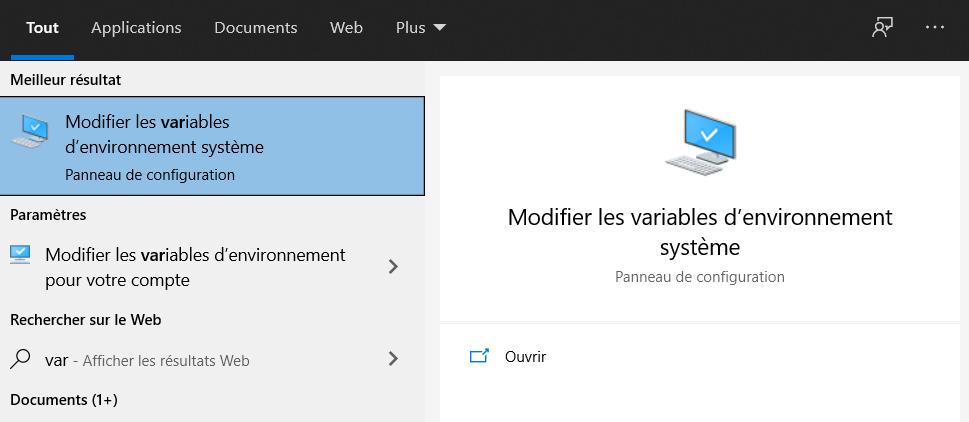
1. Télécharger Mingw 64

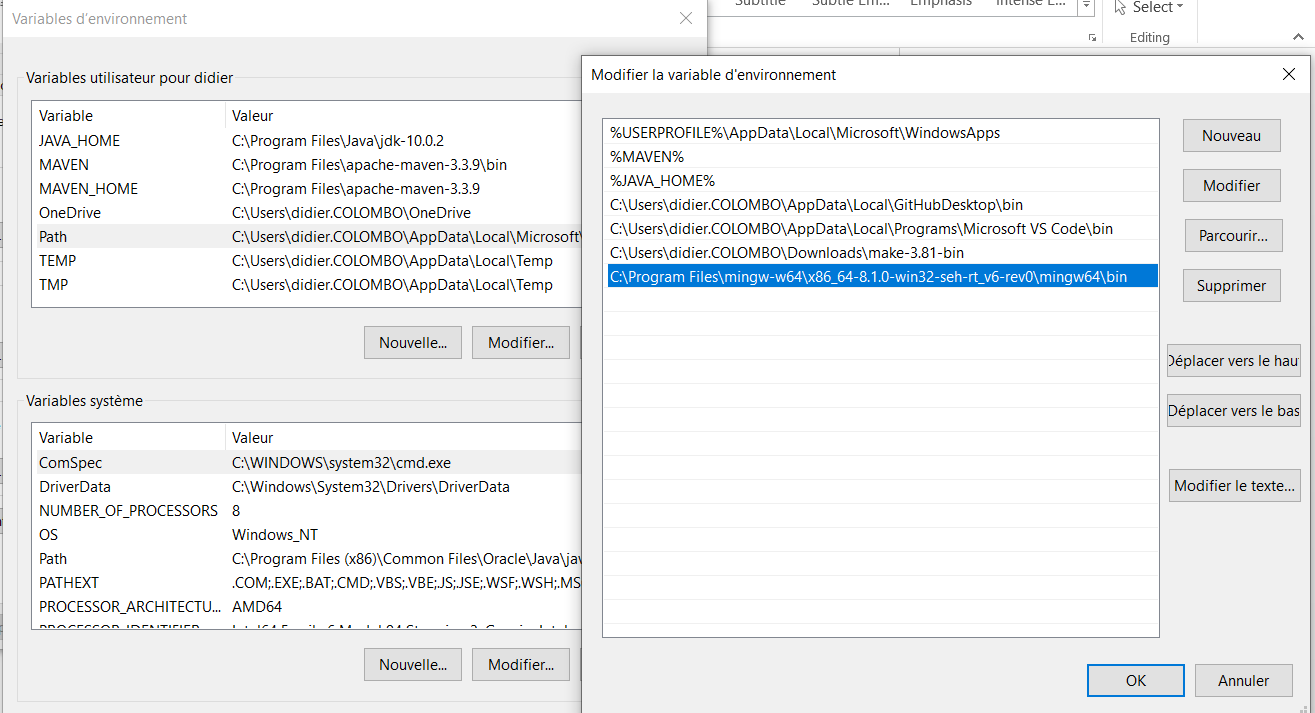
<https://sourceforge.net/projects/mingw-w64/files/Toolchains%20targetting%20Win32/Personal%20Builds/mingw-builds/installer/mingw-w64-install.exe/download>

Et installer le.

Une fois l’installation terminé, vous devriez avoir quelque chose du genre: C:\Program Files\mingw-w64\x86\_64-8.1.0-win32-seh-rt\_v6-rev0\mingw64\bin

Aller dans vos variables d’environnement :



Puis sélectionner la variable Path, pour ajouter une nouvelle variable d’environnement comme ci-dessous: 

Une fois terminer, redémarrer votre pc pour que l’installation soit prête.

Vous pourrez alors lancer eclipse.