**Fonction retourne le numero du jour de la semaine qui est le jour du 1er janvier**

**//1 ANNEE bisextile ?**

**Si Oui**

**cout << « lu ma me je ve sa di» ;**

**Si Non**

**affichage.cpp**

Fct : Affichage message bienvenue et « manuel » du programme

+++++++++++++++++++++++++++++++++++++++++++++++++++++

void msg\_bienvenue\_explications(void)

{

cout << « Bonjour, bienvenue dans notre calculateur de calendrier » << endl ;

cout << « Saisissez l’annee dont vous souhaitez afficher le calendrier » << endl

<< «  Veuillez entrer une valeur situee entre 1800 et 2100 » << endl ;

}

Fct : Affichage message d’erreur

+++++++++++++++++++++++++++

Retourne rien…. À discuter.

void msg\_erreur\_saisie() {

cout << “Erreur, veuillez saisir une autre date entre 1800 et 2100. » << endl;

}

Fct : Affichage message menu quitter ou relancer le programme ?

+++++++++++++++++++++++++++++++++++++++++++++++++++++

void msg\_menu\_quitter\_prg(void) {

cou t << « Voulez-vous afficher un autre calendrier ? » << endl ;

<< « [ O / N ] ? » << endl ;

}

**saisie.cpp**

Fct : Saisie utilisateur de l’annee souhaitee pour le calendrier

+++++++++++++++++++++++++++++++++++++++++++++++++++

int saisie\_annee\_calendrier(const int& annee)

{

bool controle\_flux\_saisie ;

do {

controle\_flux\_saisie = not(cin >> annee) ;

if (controle\_flux\_saisie) {

nettoyage\_flux();

msg\_erreur\_saisie()

}

VIDER\_BUFFER ;

} while (annee < 1800 || annee > 2100) ;

return EXIT\_SUCESS ;

}

Fct : Nettoyage de flux

+++++++++++++++++++

Si nettoyage a fonctionné fonction retourne EXIT\_SUCCESS, sinon, EXIT\_FAILURE

bool nettoyage\_flux(void)

{

If (cin.clear())

return EXIT\_SUCCESS ;

return EXIT\_FAILURE;

}

Fct : Menu quitter recommencer

++++++++++++++++++++++++++++

Retourne EXIT\_SUCCESS si l’utilisateur souhaite rentrer une nouvelle date, sinon, si il désire quitter retourne EXIT\_FALURE

bool menu\_quitter(void) {

bool quitte\_oui\_non ;

char c;

msg\_menu\_quitter\_prg() ;

do {

cin >> c ;

if (c == ‘o’ || c == ‘O’)

return EXIT\_SUCCESS ;

else if (c == ‘n’ || c == ‘N’)

return EXIT\_FAILURE ;

nettoyage\_flux();

msg\_erreur\_saisie() }

} while (c != ‘o’ || c != ‘O’ || c != ‘n’ || c != ‘N’) ;

}

//Deux option pour la boucle quitter\_ou\_non

Soit un return EXIT\_SUCESS qui casse le while(1)

Soit un booléan

Plan v1.2

Deadline : Vendredi 4 novembre 2022, 23h

Choix de la langue : français, sans accents

**MISE EN PLACE, REPARTITION DES TACHES +++++++++++++++++++++++++++++++++++++++++++++++++**

Fichiers à créer pour le projet :

* [x] .cpp, .hpp
* [x] cMake avec les includes
* [x] dossier github
* [ ] création de 2 branches distinctes Julia et Nicolas
* [x] plan – pseudo – code
* [x] schéma mindmap des fichiers et relations entre eux

Liste des sous-programmes à créer :

* [] Fct1 : Saisie utilisateur de l’année souhaitée pour le calendrier  : Julia
* [] Fct2.1 : Vérification saisie utilisateur : Julia
* [ ] Fct2.2 Affichage message erreur
* [] Fct3 : déterminer Jour du premier 1er janvier avec ctime : Nicolas
* [] Fct4 : Année bissextile ou non ? : Nicolas
* [] Fct5 : Affichage du calendrier : Nicolas, Julia
  + Afficher nom du mois, Année
  + Afficher L M M J V S D
* [] Fct6 : Menu : quitter, recommencer : Julia

Liste des erreurs à tester :

* Saisir qqchose hors de intervalle [1800 –2100]
* Saisir qqchose qui n’est pas un int

**VERIFICATION POUR LE RENDU** +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

Fichiers pour le rendu :

* [ ] .cpp
* [ ] .hpp
* [ ] pdf

Vérifications avant le rendu :

* [ ] Noms de fichiers
* [ ] En – tête pour chaque fichier
* [ ] Commentaires judicieux, sans accents, français
* [ ] Nom des variables
* [ ] Alignement du code, lisibilité
* [ ] Warnings
* [ ] Opérations pas autorisées