

```

1  /*
2  -----
3  Fichier      : calendrier.cpp
4  Nom du labo  : Laboratoire 03
5  Auteur(s)   : Flavio Sovilla, Nicolas Benitez
6  Date        : 26.10.2020
7  But         : Permettre à l'utilisateur d'entrer une date entre 1900
8                et 2100. Le programme va par la suite ressortir le
9                calendrier de l'année avec tous les mois et les dates
10               des différents jours.
11
12  Remarque(s) : Les valeurs de verifications d'une année bissextile ont été laissées
13                en dur dans le code puisqu'elles proviennent d'une formule. C'est aussi
14                le cas pour la formule du calcul du bon jour du mois de janvier.
15
16                Source pour calcul du jour du 1er Janvier :
17                https://cs.uwaterloo.ca/~alopez-o/math-faq/node73.html
18
19                Source pour calcul du nombre de jours dans 1 mois :
20                http://www.dispersiondesign.com/articles/time/number\_of\_days\_in\_a\_month
21
22  Compilateur : MinGW-w64
23  -----
24  */
25  #include <cstdlib> // Pour le EXIT_SUCCESS
26  #include <iostream> // Pour le flux
27  #include <limits> // Pour le numeric_limits<streamsize>
28  #include <iomanip> // Pour le setw(...)
29  #include <cmath> // Pour le calcul du jour du 1er Janvier -> fonction floor()
30
31  using namespace std;
32
33  #define VIDER_BUFFER cin.ignore(numeric_limits<streamsize>::max(), '\n')
34
35  int main() {
36
37      // -----
38      // Variables constantes
39      const int ESPACE_JOURS = 2,
40                ESPACE_ALIGNEMENT = 3,
41                NBRE_COLONNES = 7,
42                ANNEE_MIN = 1900,
43                ANNEE_MAX = 2100,
44                MOIS_DEPART = 1,
45                MOIS_FIN = 12;
46
47      // -----
48      // Variables générales
49      enum class Mois {
50          JANVIER = 1,
51          FEVRIER,
52          MARS,
53          AVRIL,
54          MAI,
55          JUIN,
56          JUILLET,
57          AOUT,
58          SEPTEMBRE,
59          OCTOBRE,
60          NOVEMBRE,
61          DECEMBRE
62      };
63
64      char choixUtilisateur;
65
66      // -----
67      // Message de bienvenue
68      cout << "ce programme ..." << endl;
69
70      // -----
71      // Boucle de recommencement selon entrée utilisateur
72      do{
73
74          // -----
75          // Variables utilisées pour gérer l'entrée de l'utilisateur
76          int annee;
77

```

```

78     bool verifAnnee = true,
79         bissextile = true;
80
81     // -----
82     // Saisie de l'année par l'utilisateur
83     do {
84         cout << "entrer une valeur [" << ANNEE_MIN << "-" << ANNEE_MAX << "] : ";
85         cin >> annee;
86
87         if (cin.fail()) {
88             cin.clear();
89         }
90         VIDER_BUFFER;
91
92         if (annee >= ANNEE_MIN && annee <= ANNEE_MAX) {
93             verifAnnee = false;
94         }
95         else {
96             cout << "//!\\ recommencer" << endl;
97         }
98     } while (verifAnnee);
99     cout << endl;
100
101     // -----
102     // Vérification de l'année bissextile
103     // Ne pas modifier les valeurs (voir remarques) !
104     bissextile = (annee % 400 == 0 || (annee % 4 == 0 && annee % 100 != 0));
105
106     // -----
107     // Formule pour calculer le jour de la semaine pour le 1er janvier
108     // Ne pas modifier les valeurs (voir remarques) !
109     int anneeSiecle = (annee % 100) - 1 ; // les deux derniers chiffres de l'annee
110     int siecle      = annee / 100;      // les deux premiers chiffres de l'annee
111     int jourSemaine =
112         + (int)floor((2.6 * 11) - 0.2)
113         + (int)floor(anneeSiecle / 4)
114         + (int)floor(siecle / 4)
115         - 2 * siecle + anneeSiecle
116         + 1 ;
117
118     // Un decalage s'opere sur le 1er Janvier puisque le resultat du calcul precedent
119     // se base sur une semaine commençant un dimanche
120     int jourDepart =
121         (jourSemaine % 7) == 0 ?
122         (jourSemaine % 7) + 6 :
123         (jourSemaine % 7) - 1;
124
125     // -----
126     // Initialisation de la boucle des calendriers
127     for (int i = MOIS_DEPART; i <= MOIS_FIN; ++i) {
128
129         int nbreJour = 1;
130
131         switch(i) {
132             case (int)Mois::JANVIER:    cout << "JANVIER ";
133                                         break;
134             case (int)Mois::FEVRIER:    cout << "FEVRIER ";
135                                         break;
136             case (int)Mois::MARS:        cout << "MARS ";
137                                         break;
138             case (int)Mois::AVRIL:      cout << "AVRIL ";
139                                         break;
140             case (int)Mois::MAI:         cout << "MAI ";
141                                         break;
142             case (int)Mois::JUIN:        cout << "JUIN ";
143                                         break;
144             case (int)Mois::JUILLET:     cout << "JUILLET ";
145                                         break;
146             case (int)Mois::AOUT:        cout << "AOUT ";
147                                         break;
148             case (int)Mois::SEPTEMBRE:   cout << "SEPTEMBRE ";
149                                         break;
150             case (int)Mois::OCTOBRE:     cout << "OCTOBRE ";
151                                         break;
152             case (int)Mois::NOVEMBRE:    cout << "NOVEMBRE ";
153                                         break;
154             case (int)Mois::DECEMBRE:    cout << "DECEMBRE ";

```

```

155                                     break;
156         default: break;
157     }
158
159     // -----
160     // Calcul du nombre de jour d'un mois
161     // On remarque un pattern qui se répète tous les 7 mois, dans lequel se trouve
162     // un deuxième pattern se repetant tous les 2 mois. Il ne reste plus qu'a gerer
163     // le mois de fevrier et les annees bissextiles. Cela donne le calcul suivant
164     int nbreJoursMois = (i == 2) ? (28 + bissextile) : 31 - (i - 1) % 7 % 2;
165
166     // -----
167     // Affichage des colonnes avec gestion des espaces
168     cout << annee << endl;
169     cout << " L M M J V S D" << endl;
170
171     int sautJours = 0;
172
173     while(jourDepart--) {
174         cout << setw(ESPACE_ALIGNEMENT) << " ";
175
176         ++sautJours;
177
178         if (sautJours % NBRE_COLONNES == 0) {
179             cout << endl;
180         }
181     }
182
183     // -----
184     // Affichage des jours du mois
185     while(nbreJour <= nbreJoursMois){
186         cout << setw(ESPACE_JOURS) << nbreJour << " ";
187
188         ++nbreJour;
189         ++sautJours;
190
191         if (sautJours % NBRE_COLONNES == 0) {
192             cout << endl;
193         }
194     }
195     cout << endl;
196
197     // On recalcule le jour de départ pour le mois suivant
198     jourDepart = sautJours % NBRE_COLONNES;
199
200     // On met un saut de ligne s'il n'y en a pas déjà un
201     if (jourDepart) {
202         cout << endl;
203     }
204 }
205
206 // -----
207 // Invitation à recommencer avec gestion de l'entree utilisateur
208 do{
209     cout << "Voulez-vous recommencer [O/N] ? ";
210     cin >> choixUtilisateur;
211
212     if (cin.fail()) {
213         cin.clear();
214     }
215     VIDER_BUFFER;
216 }while(choixUtilisateur != 'N' && choixUtilisateur != 'O');
217
218 } while(choixUtilisateur == 'O');
219
220 return EXIT_SUCCESS;
221
222 }
```