

```

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.
14
15                Source pour calcul du jour du 1er Janvier :
16                https://cs.uwaterloo.ca/~alopez-o/math-faq/node73.html
17
18                Source pour calcul du nombre de jours dans 1 mois :
19                http://www.dispersiondesign.com/articles/time/number\_of\_days\_in\_a\_month
20
21  Compilateur : MinGW-w64
22  -----
23  */
24  #include <cstdlib>    // Pour le EXIT_SUCCESS
25  #include <iostream>  // Pour le flux
26  #include <limits>    // Pour le numeric_limits<streamsize>
27  #include <iomanip>    // Pour le setw(...)
28  #include <cmath>     // Pour le calcul du jour du 1er Janvier -> fonction floor()
29
30  using namespace std;
31
32  #define VIDER_BUFFER cin.ignore(numeric_limits<streamsize>::max(), '\n')
33
34  int main() {
35
36      // -----
37      // Variables constantes
38      const int ESPACE_JOURS      = 2,
39                ESPACE_ALIGNEMENT = 3,
40                NBRE_COLONNES     = 7,
41                ANNEE_MIN         = 1900,
42                ANNEE_MAX         = 2100,
43                MOIS_DEPART       = 1,
44                MOIS_FIN          = 12;
45
46      // -----
47      // Variables générales
48      enum class Mois {
49          JANVIER = 1,
50          FEVRIER,
51          MARS,
52          AVRIL,
53          MAI,
54          JUIN,
55          JUILLET,
56          AOUT,
57          SEPTEMBRE,
58          OCTOBRE,
59          NOVEMBRE,
60          DECEMBRE
61      };
62
63      char choixUtilisateur;
64
65      // -----
66      // Message de bienvenue
67      cout << "ce programme ..." << endl;
68
69      // -----
70      // Boucle de recommencement selon entrée utilisateur
71      do{
72
73          // -----
74          // Variables utilisées pour gérer l'entrée de l'utilisateur
75          int annee;
76
77          bool verifAnnee = true,

```

```

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

```

```

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

```