

1 PROJET : QUEL JOUR SERA-T-ON DEMAIN ?

1.1 PROBLÉMATIQUE

Si aujourd'hui nous sommes le 9 octobre 2012, nous serons demain le 10 10 2012.
Mais quel jour sera-t-on demain dans tous les cas ?

1.2 PRÉ-REQUIS

Avoir réaliser les activités de la Proglet **abcdAlgos** de Java'scool, c'est à dire :

- Utiliser des séquences d'instructions.
- Apprendre à se servir de variables.
- Découvrir l'instruction conditionnelle.
- Se familiariser avec les fonctions.
- Programmer avec des boucles.

Pour aller plus loin on pourra lire la page : <http://javascool.gforge.inria.fr/documents/sujets-mathinfo/sujet-about-tableaux.htm> qui présente les tableaux.

1.3 CAPACITÉS NOUVELLES OU APPROFONDIES

Savoirs	Capacités
Algorithmes simples	Comprendre un algorithme et expliquer ce qu'il fait. Modifier un algorithme existant pour obtenir un résultat différent. Concevoir un algorithme. Programmer un algorithme.
Types de données	Choisir un type de donnée en fonction d'un problème à résoudre. nombre entier, booléen , tableau
Fonctions - notion ; - portée des variables ;	Concevoir l'entête (ou l'interface) d'une fonction, puis la fonction elle-même.

1.4 ORGANISATION/DURÉE

L'activité est réalisée par groupe de 3 ou 4 sur deux séances

Vous devez vous répartir les différentes activités.

Un groupe présentera son travail la semaine suivante, une présentation informatique est requise.

Remarque :: le groupe sera désigné lors de la deuxième séance en fonction de l'avancée des travaux de chaque groupe.

1.5 ACTIVITÉS

1.5.1 NOMBRE DE JOUR DU MOIS

La durée d'un mois est variable, écrire un programme (une fonction) qui à partir du numéro du mois retourne la durée du mois en jour.

Dans cette première activité, on ne prendra pas en compte les années bissextiles (février=28 jours).

1.5.2 ANNÉE BISSEXTILE

Une année est une année bissextile si c'est une année multiple de 4 (1996, 2004,...) mais pas de 100 (1800, 1900, ...) sauf si c'est de 400 (1600, 2000,...).

Écrire une fonction qui retourne le nombre de jour du mois de février en fonction de l'année.

Écrire une autre version qui elle retourne un booléen (vrai/true, faux/false)

1.5.3 NOMBRE DE JOUR DU MOIS

Reprendre le programme donnant le nombre de jour du mois pour prendre en compte les années bissextiles.

1.5.4 INTERFACE UTILISATEUR

Écrire un programme qui demande à l'utilisateur, le jour, le mois, l'année et qui vérifie que cette date est valide, qui repose la question dans le cas contraire.

1.5.5 DÉTERMINATION DE LA DATE DU JOUR SUIVANT

Écrire la fonction qui détermine la date du jour suivant la date rentrée par l'utilisateur et qui l'affiche.

1.5.6 SYNTHÈSE

À partir des activités précédentes, écrire le programme qui détermine la date du jour suivant.

2 CORRECTION

2.1 SOLUTION 1

Dans cette première solution tous les fonctions sont réalisées dans le programmes principal. le programmes est peu lisible, cette solution est à éviter.

Ouvrir le fichier : JD1-ToutEnVrac.jvs.

2.1.1 LISTING

```
1 void main(){
2 int jj=0, mm=0, aa=0; // jour, mois et annee
3 int jjsuivant, mmsuivant, aasuivante;
4 int nbJourDuMois=31;
5
6 // dialogue utilisateur
7 print("jour=?");
8 jj=readInteger();
9 println(jj);
10 print("mois=?");
11 mm=readInteger();
12 println(mm);
13 print("annee=?");
14 aa=readInteger();
15 println(aa);
16
17 // verification coherence de la date tapee
18 // d'abord l'annee
19 while (aa<1582){
20     println("Le calendrier gregorien a ete impose en France a partir du 9 decembre 1582");
21     println("Rentrer une annee possible");
22     aa=readInt();
23 }
24 // puis le mois
25 while (mm<1 || mm>12){
26     println("taper un mois compris entre 1 et 12");
27     mm=readInt();
28 }
29 // le jour (le jour demandee ne doit pas etre au dela u dernier jour du mois(mm))
30
31     if (mm==1||mm==3||mm==5||mm==7 ||mm==8 || mm==10||mm==12){
32         nbJourDuMois=31;
33     }else if (mm==4||mm==6||mm==9 ||mm==11 ){
34         nbJourDuMois= 30;
35     }else{
36         if (aa%400==0){
37             nbJourDuMois= 29;
38         }else if(aa%4==0){
39             if (aa%100!=0){
40                 nbJourDuMois= 29;
41             }else{ nbJourDuMois= 28;}
42         }else{ nbJourDuMois= 28;}
43     }
44
45
46 while (jj<1 || jj>nbJourDuMois){
47     println("taper un numero du jour compriss entre 1 et "+ nbJourDuMois);
48     jj=readInt();
49 }
50
51 println("La date demandee est le jour suivant du "+jj+"/"+mm+"/"+aa);
```

```

52 // fin dialogue
53
54 if (jj < nbJourDuMois) {
55     jjsuivant = jj + 1;
56     mmsuivant = mm;
57     aasuivante = aa;
58 } else {
59     jjsuivant = 1;
60     mmsuivant = mm + 1;
61     aasuivante = aa;
62     if (mmsuivant == 13) {
63         mmsuivant = 1;
64         aasuivante = aa + 1;
65     }
66 }
67 print("Je reflechi.");
68 for (int i = 1; i < 10; i = i + 1) {
69     sleep(100);
70     print(". ");
71 }
72 println(" C'est le " + jjsuivant + "/" + mmsuivant + "/" + aasuivante);
73 }

```

Remarque : Pour la suite, on ne précisera plus la partie comprise relative au dialogue utilisateur sauf s'il y a changement.

2.1.2 POINTS CLÉS

variables : int

entrées/sorties : ReadInteger(), print(...),
println(...),...

structure conditionnelle :

– Si Alors :

```

1 if (condition booléenne) {
2     instructions;
3 }

```

– Si Alors Sinon

```

1 if (condition booléenne) {
2     instructions;
3 } else {
4     instructions;
5 }

```

– Si Alors... Sinon Si...Sinon

```

1 if (condition booléenne) {
2     instructions;
3 } else if {
4     instructions;
5 } else {
6     instructions;
7 }

```

boucle TANT QUE :

```

1 while (condition booléenne) {
2     instructions;
3 }

```

opérateurs logiques et mathématiques

- ET : &&
- OU : ||
- NON : !
- modulo : %

2.2 SOLUTION 2

Pour rendre le programme plus lisible, on peut créer une fonction qui détermine le nombre de jour et qui retourne la valeur dans le programme principal.

Ouvrir le fichier : JD2-fonction-nbJourMois.jsv.

2.2.1 LISTING DU PROGRAMME PRINCIPAL

```
1 void main(){
2 int jj=0, mm=0, aa=0; // jour, mois et annee
3 int jjsuivant, mmsuivant, aasuivante;
4 // dialogue utilisateur
5
6 // Voir plus haut
7
8 // fin dialogue utilisateur
9
10 if(jj<nbJourDuMois(mm,aa)){
11     jjsuivant=jj+1;
12     mmsuivant=mm;
13     aasuivante=aa;
14 }else {
15     jjsuivant=1;
16     mmsuivant=mm+1;
17     aasuivante=aa;
18     if (mmsuivant==13){
19         mmsuivant=1;
20         aasuivante=aa+1;
21     }
22 }
23 print("Je reflechi !");
24 for (int i=1;i<10;i=i+1){
25     sleep(100);
26     print(". ");
27 }
28 println(" C'est le "+jjsuivant+"/"+mmsuivant+"/"+aasuivante);
29
30 }
```

2.2.2 LISTING DE LA FONCTION nbJourDuMois(int numMois, int annee)

```
1
2 // Fonction nbJourDuMois(int numMois, int annee), retourne le nombre de jour (entier) du mois
3 // la fonction est ici réalisée par des tests successifs
4 int nbJourDuMois(int numMois, int annee){
5     if (numMois==1||numMois==3||numMois==5||numMois==7
6         ||numMois==8 || numMois==10||numMois==12){
7         return 31;
8     }else if (numMois==4||numMois==4||numMois==9 ||numMois==11 ){
9         return 30;
10    }else{
11        if (annee%400==0){
12            return 29;
13        }else if(annee%4==0){
14            if (annee%100!=0){
15                return 29;
16            }else{ return 28;}
17        }else{ return 28;}
18    }
19 }
```

2.2.3 POINTS CLÉS

On complète les points précédents avec :

variables : les variables booléennes `boolean`;

fonctions :

– création de la fonction

```
1 type nomDeLaFonction(type variable, \dots){
2 corps de la fonction;
3 instruction;
4 return valeur a retourner;
5 }
```

– utilisation de la fonction : `nomDeLaFonction(var)`

boucle POUR

```
1 for (int i=1;i<10;i=i+1){
2 sleep(100);
3 print(". ");
4 }
```

2.3 SOLUTION 3

Une évolution de la solution précédente consiste à extraire de la fonction `nbJourDuMois(int numMois, int annee)` celle qui détermine si l'année est bissextile.

Ouvrir le fichier : `JD3-fonctions-nbJourMois-Bissextile.js`

- `nbJourDuMois(int numMois, int annee)` : cette fonction retourne le nombre de jour (int) du mois, en fonction du numéro du mois et de l'année (pour février).
- `bissextile (int x)` : cette fonction est appelée par la précédente, elle détermine si l'année est bissextile ou non. Elle retourne un booléen.

2.3.1 LISTING DU PROGRAMME PRINCIPAL

Le programme principal ne change pas par rapport au précédent.

2.3.2 LISTING DE LA FONCTION `nbJourDuMois(int numMois, int annee)`"

```
1 int nbJourDuMois(int numMois, int annee){
2     if (numMois==1||numMois==3||numMois==5||numMois==7
3         ||numMois==8 || numMois==10||numMois==12){
4         return 31;
5     }else if (numMois==4||numMois==4||numMois==9 ||numMois==11 ){
6         return 30;
7     }else{
8         if (bissextile(annee)){// appel de la fonction bissextile avec le n de l'année
9             return 29;
10        }else{ return 28;}
11    }
12 }
```

2.3.3 LISTING DE LA FONCTION `bissextile (int x)`"

```
1 boolean bissextile (int x){
2     if (x%400==0){
3         return true;
4     }else if(x%4==0){
5         if (x%100!=0){
6             return true;
7         }else{ return false;}
8     }else{ return false;}
9 }
```

2.4 SOLUTION 4

Cette nouvelle solution, est une autre approche de la résolution, au lieu ici de faire un ensemble de test successifs pour déterminer la longueur du mois, on choisi de mémoriser le résultat une fois pour toute, sauf pour le mois de février qu'il faudra calculer.

Pour cela, nous allons utiliser un nouveau type de variable, les tableaux, celui-ci est initialisée avec les valeurs par défaut :

```
1 int nbJourDuMois []={31,28,31,30,31,30,31,31,30,31,30,31};
```

.

Pour lire une valeur dans le tableau il suffit de l'appeler par

```
1 nbJourDuMois [3];
```

. Attention cet appel retourne la 4^e valeur, les tableaux commençant à l'indice 0.

Pour écrire une valeur dans le tableau, il suffit d'écrire :

```
1 nbJourDuMois [1]=28;
```

.

Ouvrir le fichier : JD4-MoisEnTableau-Bissextile.java

2.4.1 LISTING DU PROGRAMME PRINCIPAL

```
1 void main(){
2 int aa=0, mm=0, jj=0;
3 int jjsuivant, mmsuivant, aasuivante;
4 int nbJourDuMois []={31,28,31,30,31,30,31,31,30,31,30,31}; // tableau de mois
5 // attention les tableaux commencent a l'indice 0
6 // dialogue utilisateur
7
8 print("jour=?");
9 jj=readInteger();
10 println(jj);
11 print("mois=?");
12 mm=readInteger();
13 println(mm);
14 print("annee=?");
15 aa=readInteger();
16 println(aa);
17
18 // verification coherence de la date tapee
19 // d'abord l'annee
20 while (aa<1582){
21     print("Le calendrier gregorien a ete impose en France a partir du 9 decembre 1582.");
22     print("Rentrer une annee possible, Annee=?");
23     aa=readInt();
24     println(aa);
25 }
26 // puis le mois
27 // test logique pour verifier la validite du mois dans une boucle while
28
29 while (mm<1 || mm>12){
30     println("taper un mois compris entre 1 et 12. Mois=?");
31     mm=readInt();
32 }
33
34 // calcul au prealable de la longueur du mois de fevrier
35
36     nbJourDuMois[2-1]=bissextile(aa); // attention les tableaux commencent a l'indice 0
37
38 /*le jour (le jour demandee ne doit pas etre au dela du dernier jour du mois(mm)
```

```

39  * lecture de la longueur du mois dans le tableau
40  */
41  while (jj<1 || jj>nbJourDuMois[mm-1]){
42      print("taper un numero du jour compris entre 1 et "+ nbJourDuMois[mm-1] + ". Jour=?");
43      jj=readInt();
44      println(jj);
45  }
46
47  print("La date demandee est le jour suivant du "+jj+"/"+mm+"/"+aa);
48
49  if(jj<nbJourDuMois[mm-1]){ // mm varie de 1 Ã 12 et le tableau de 0 Ã 11
50      jjsuivant=jj+1;
51      mmsuivant=mm;
52      aasuivante=aa;
53  }else {
54      jjsuivant=1;
55      mmsuivant=mm+1;
56      aasuivante=aa;
57      if (mmsuivant==13){
58          mmsuivant=1;
59          aasuivante=aa+1;
60      }
61  }
62  println(" soit le "+jjsuivant+"/"+mmsuivant+"/"+aasuivante);
63  }
64
65  }

```

2.4.2 LISTING DE LA FONCTION bissextile(int annee)

La fonction bissextile retourne la longueur du mois de février ; On remarquera l'écriture de la condition logique de la structure conditionnelle.

```

1  int bissextile(int annee){
2      /* la condition (aa % 4==0&&aa%100>0 )|| (aa%400==0) est vrai
3      * si l'annee est bissextile.
4      * aa est une variable locale, mÃame si elle un nom identique a celle
5      * definie a l'exterieur, elle n'existe que dans la fonction.
6      */
7      int aa=annee;
8      if ((aa % 4==0&&aa%100>0 )|| (aa%400==0)){
9          return 29; // attention les tableaux commencent a l'indice 0
10     }else{
11         return 28;
12     }
13
14 }

```

2.4.3 POINTS CLÉS

Variables :

Les variables de type tableau, comprendre la création et l'utilisation d'un tableau.

Tests logiques : Comprendre le test logique (aa % 4==0&&aa%100>0)|| (aa%400==0)

2.5 SOLUTION 5

Cette solution et la suivante prolonge l'idée de décomposer un programme « complexe » en fonctions plus simples et plus faciles à maintenir. Effectivement on peut imaginer un programme qui ne comporterait que deux fonctions :

- demanderLaDate(); qui réalise le dialogue avec l'utilisateur et qui permet d'obtenir une date valide.
- calculerLaDate(jj, mm, aa); qui effectue le calcul du jour suivant puis retourne la date à afficher.

Il est nécessaire pour que cela fonctionne de définir comment transmettre les résultats de la première fonction à la seconde. Cette première solution utilise les variables globales, c'est à dire des variables utilisables par toutes les fonctions du programme. Chaque fonction peut donc, lire ou écrire ces variables.

Ouvrir le fichier :JD5-VariablesGlobales.jsv

2.5.1 LISTING DU PROGRAMME PRINCIPAL

```
1
2 static int aa = 0, mm = 0, jj = 0;
3
4 static int jjsuivant, mmsuivant, aasuivante;
5
6 /* les variables declarees avec static sont globales, c'est necessaire ici
7  * car les deux fonctions void demanderLaDate() et void calculerLaDate()
8  * ne retourne pas une seule valeur mais plusieurs ce n'est pas forcément ici la
9  * meilleure solution.
10 */
11 void main() {
12     demanderLaDate();
13     // fonction de dialogue utilisateur pour obtenir une date valable
14     calculerLaDate(jj, mm, aa);
15     // fonction de calcul de la date
16     println(" soit le " + jjsuivant + "/" + mmsuivant + "/" + aasuivante);
17     // affichage du resultat
18 }
```

Le programme principal est maintenant très court.

On constate que les variables sont définies à l'extérieur de la fonction main() et des autres fonctions, elles sont globales. Le mot clé static permet de définir ces variables.

2.5.2 LISTING DE demanderLaDate()

```
1 void demanderLaDate() {
2     //dialogue utilisateur
3     print("jour=?");
4     jj = readInteger();
5     println(jj);
6     print("mois=?");
7     mm = readInteger();
8     println(mm);
9     print("annee=?");
10    aa = readInteger();
11    println(aa);
12    // verification coherence de la date tapee
13    // d'abord l'annee
14    while (aa < 1582) {
15        println("Le calendrier gregorien a ete impose en France a partir du 9 decembre 1582");
16        println("Rentrer une annee possible");
17        aa = readInt();
18    }
19    // puis le mois
20    while (mm < 1 || mm > 12) {
21        println("taper un mois compris entre 1 et 12");
22        mm = readInt();
23    }
24 }
```

```

23     }
24     // le jour (le jour demandee ne doit pas être au delà du dernier jour du mois(mm))
25     while (jj < 1 || jj > nbJourDuMois(mm, aa)) {
26         println("taper un numero du jour compris entre 1 et " + nbJourDuMois(mm, aa));
27         jj = readInt();
28     }
29     print("La date demandee est le jour suivant du " + jj + "/" + mm + "/" + aa);
30 }
31
32 void calculerLaDate(int jour, int mois, int annee) {
33     if (jour < nbJourDuMois(mois, annee)) {
34         // appel de la fonction nbJourDuMois(mois,annee)
35         jjsuivant = jour + 1;
36         mmsuivant = mois;
37         aasuivante = annee;
38     } else {
39         jjsuivant = 1;
40         mmsuivant = mois + 1;
41         aasuivante = annee;
42         if (mmsuivant == 13) {
43             mmsuivant = 1;
44             aasuivante = annee + 1;
45         }
46     }
47 }

```

2.5.3 LISTING DE calculerLaDate(int jour, int mois, int annee)

```

1 void calculerLaDate(int jour, int mois, int annee) {
2     if (jour < nbJourDuMois(mois, annee)) {
3         // appel de la fonction nbJourDuMois(mois,annee)
4         jjsuivant = jour + 1;
5         mmsuivant = mois;
6         aasuivante = annee;
7     } else {
8         jjsuivant = 1;
9         mmsuivant = mois + 1;
10        aasuivante = annee;
11        if (mmsuivant == 13) {
12            mmsuivant = 1;
13            aasuivante = annee + 1;
14        }
15    }
16 }

```

Les autres fonctions sont inchangées.

Les deux fonctions demanderLaDate() et calculerLaDate() ne comporte pas le mot clé return, elle ne retourne aucune valeurs dans le programme principal. Ce type de fonction doit être déclarée avec le mot clef void. On appelle aussi ces fonctions, des procédures.

2.5.4 POINTS CLÉS

Variables : Déclaration des variable globales.

Fonctions : Fonctions qui ne retourne aucune valeur (procédure), déclarées avec le mot clé void.

2.6 SOLUTION 6

La solution avec les variables globales, est rarement une solution conseillée. Cette solution autorise toutes les autres fonctions à modifier la valeur d'une variable, en cas de modification « involontaire » d'une variable dans une fonction, il sera difficile de trouver l'origine du problème parmi toutes les fonctions. Pour cela, on préférera n'utiliser que des variables locales qui ne peuvent être modifiées que par la fonction qui les utilise. Pour passer les variables d'une fonction à une autre, on choisira de les passer par les paramètres de la fonction et nous utiliserons return pour les récupérer.

Cette dernière solution reprend l'idée de la précédente en définissant deux fonctions principales pour résoudre le problème mais la date du jour est passée ici par les paramètres de la fonction.

La difficulté principale ici est de récupérer, non pas une seule valeur des fonctions, mais 3, le jour, le mois, la date. Une fonction ne pouvant retourner qu'une seule valeur, il a été choisi ici de mettre ces trois valeurs dans un tableau et de retourner tout le tableau.

Ouvrir le fichier : JD5-VariablesLocales.java

2.6.1 LISTING DU PROGRAMME PRINCIPAL

```
1 void main(){
2     int date[] = new int[3];
3     int dateSuivante[] = new int[3];
4     int jjsuivant, mmsuivant, aasuivante;
5     date = demanderLaDate();
6     /*fonction de dialogue utilisateur pour obtenir une date valable
7     * la fonction retourne un tableau de 3 valeurs, le jour, le mois, l'année
8     */
9     dateSuivante = calculerLaDate(date);
10    /* fonction de calcul de la date
11    * la fonction prend comme paramètre un tableau de 3 valeurs (le jour, le mois, l'année)
12    * et retourne un tableau de 3 valeurs
13    */
14    jjsuivant = dateSuivante[0];
15    mmsuivant = dateSuivante[1];
16    aasuivante = dateSuivante[2];
17    println(" soit le " + jjsuivant + "/" + mmsuivant + "/" + aasuivante); // affichage du résultat
18 }
```

La date du jour et la date suivante sont déclarées dans deux tableaux.

2.6.2 LISTING DE demanderLaDate();

```
1 int[] demanderLaDate(){
2     /*La déclaration de la fonction est de type tableau d'entier
3     * int[] demanderLaDate()
4     */
5     int laDateDuJour[] = new int[3]; // déclaration du tableau à retourner
6     int aa=0, mm=0, jj=0; // paramètres internes pour faciliter la lecture
7     //dialogue utilisateur
8     print(" jour=?");
9     jj = readInteger();
10    println(jj);
11    print(" mois=?");
12    mm = readInteger();
13    println(mm);
14    print(" année=?");
15    aa = readInteger();
16    println(aa);
17    // vérification cohérence de la date tapée
18    // d'abord l'année
19    while (aa < 1582){
20        println("Le calendrier grégorien a été imposé en France à partir du 9 décembre 1582.");
21        println("Rentrer une année possible");
22    }
```

```

22     aa=readInt();
23 }
24 // puis le mois
25 while (mm<1 || mm>12){
26     println("taper un mois compris entre 1 et 12");
27     mm=readInt();
28 }
29 // le jour (le jour demandee ne doit pas Ãatre au dela du dernier jour du mois(mm))
30
31 while (jj<1 || jj>nbJourDuMois(mm,aa)){
32     println("taper un numero du jour compris entre 1 et "+ nbJourDuMois(mm,aa));
33     jj=readInt();
34 }
35 print("La date demandee est le jour suivant du "+jj+"/"+mm+"/"+aa);
36 // remplissage du tableau Å retourner
37 laDateDuJour[0]= jj;
38 laDateDuJour[1]= mm;
39 laDateDuJour[2]= aa;
40 return laDateDuJour;//retour du tableau
41 }

```

2.6.3 LISTING DE

```

1 int[] calculerLaDate(int ladededemain[]){
2     int jour= ladededemain[0];
3     int mois= ladededemain[1];
4     int annee= ladededemain[2];
5     int ladededemain[]= new int[3];
6     int jjsuiv, mmsuiv, aasuiv;
7     if (jour<nbJourDuMois(mois,annee)){// appel de la fonction nbJourDuMois(mois,annee)
8         jjsuiv=jour+1;
9         mmsuiv=mois;
10        aasuiv=annee;
11    }else {
12        jjsuiv=1;
13        mmsuiv=mois+1;
14        aasuiv=annee;
15        if (mmsuiv==13){
16            mmsuiv=1;
17            aasuiv=annee+1;
18        }
19    }
20    println(jjsuiv+", "+mmsuiv+", "+aasuiv);
21    ladededemain[0]=jjsuiv;
22    ladededemain[1]=mmsuiv;
23    ladededemain[2]=aasuiv;
24    return ladededemain;
25 }

```

2.6.4 POINTS CLÉS

Variables : Comprendre la différence entre variables globales et variables locales

Fonctions : déclaration d'une fonction qui retourne un tableau