

## Exercices du chapitre 4

### Sommaire

#### Exercices

01-*-Simulation d'instructions conditionnelles.....	2
02-*-Conditionnelles et séquence .....	3
03-**-Prix d'une place de cinéma .....	3
04-**-Calcul d'âge .....	3
05-**-Affranchissement.....	4
06-***-Calcul d'une facture d'électricité .....	4

#### Corrigés

01-*-Simulation d'instructions conditionnelles.....	2
02-*-Conditionnelles et séquence .....	3
03-**-Prix d'une place de cinéma .....	4
04-**-Calcul d'âge .....	6
05-**-Affranchissement.....	7
06-***-Calcul d'une facture d'électricité .....	8

## 01-\*-Simulation d'instructions conditionnelles

Réaliser **plusieurs simulations** du fonctionnement des algorithmes suivants, en choisissant chaque fois une valeur différente pour HEURE (jeux d'essai). Les choix successifs de la valeur de HEURE doivent tester le fonctionnement de chaque partie de l'algorithme. Prendre les mêmes valeurs pour tester les deux algorithmes test\_1 et test\_2.

**algorithme** test\_1

*/\* Déclarations \*/*

**Variables**

heure : entier

**Début**

écrire ("Heure entre 0 et 23")

lire (heure)

si heure > 12 alors

    écrire ("soir")

sinon

    écrire ("matin')

finsi

**Fin**

**algorithme** test\_2

*/\* Déclarations \*/*

**Variables**

heure : entier

**Début**

écrire ("Heure entre 0 et 23")

lire (heure)

si heure >= 0 ET heure <= 23  
    alors

    si heure > 12 alors

        écrire ("soir")

    sinon

        écrire ("matin')

    finsi

finsi

**Fin**

### simulation n°1

ligne	heure	écran	ligne	heure	écran
1			1		
2			2		

### simulation n°2

ligne	heure	écran	ligne	heure	écran
1			1		
2			2		

### simulation n°3

ligne	heure	écran	ligne	heure	écran
1			1		
2			2		

## 02-\* -Conditionnelles et séquence

Les deux algorithmes suivants sont-ils équivalents ? Pour vérifier la réponse réaliser trois simulations de chaque algorithme lorsque l'utilisateur donne la valeur 5, puis la valeur 0, puis la valeur -5.

**algorithme** test\_1

*/\* Déclarations \*/*

**Variables**

nb : entier

**Début**

écrire("Donner un nombre  
entier")

lire (nb)

si nb <= 0 alors

nb ← nb + 5

sinon

nb ← nb - 5

finsi

écrire("maintenant le nombre  
vaut : " , nb)

**Fin**

**algorithme** test\_2

*/\* Déclarations \*/*

**Variables**

heure : entier

**Début**

écrire("Donner un nombre  
entier")

lire (nb)

si nb <= 0 alors

nb ← nb + 5

finsi

si nb > 0 alors

nb ← nb - 5

finsi

écrire ("maintenant le nombre  
vaut : " , nb)

**Fin**

## 03-\*\*-Prix d'une place de cinéma

Il y a deux tarifs : plein (**7 Euros**) et réduit (**4,50 Euros**).

Le tarif réduit s'applique :

- à tout le monde le mercredi,
- aux étudiants quelque soit leur âge et quelque soit le jour,
- aux moins de 18 ans et plus de 65 ans SAUF les samedi et dimanche.

Ecrire un algorithme qui recueille les données nécessaires et annonce le prix à payer pour la place.

## 04-\*\*-Calcul d'âge

Ecrire un algorithme qui calcule et affiche l'âge d'une personne après lui avoir demandé, avec les variables appropriées :

- sa date de naissance
- la date du jour

L'âge est un nombre **entier** d'années complètement écoulées. On pourra, en option, souhaiter bon anniversaire.

**Remarque** : les dates sont supposées valides, la date de naissance est supposée **antérieure** à la date du jour.

On pourra compléter l'algorithme en programmant un message d'erreur si l'antériorité de la date de naissance n'est pas respectée.

## 05-\*\*-Affranchissement

Un usager de La Poste veut envoyer un courrier. On lui fournit les tarifs (partiels pour simplifier, mais authentiques). Il veut connaître, en fonction du poids et de la destination de son courrier, quel est le prix à payer pour l'envoi de celui-ci, en service **rapide** et en service **économique**. On suppose, toujours pour simplifier, que son courrier pèse moins de 100 grammes et que sa destination est la France Métropolitaine ou DOM-TOM.

**Ecrire l'algorithme** qui après avoir demandé à l'usager de fournir :

- le poids de son courrier (en grammes)
- un code (M pour Métropole, D pour DOM, ou T pour TOM)

**lui indique les deux tarifs : économique et rapide** (en fonction du poids et du code fourni).

L'utilisateur est supposé donner des valeurs respectant les conditions et les codes attendus.

Voici les tarifs.

Lettre service rapide France Métropolitaine (tarif MR) :

poids jusqu'à *	20 g	50 g	100 g
tarifs	0,55 euros	0,88 euros	1,33 euros

Lettre service économique France Métropolitaine (tarif ME) :

poids jusqu'à *	20 g	50 g	100 g
tarifs	0,50 euros	0,72 euros	0,87 euros

Lettre service rapide DOM-TOM :

poids	DOM	TOM
jusqu'à * 20 g	0,55 euros	0,55 euros
plus de 20 g	Tarif MR + 0,06 euros par 10 g **	Tarif MR + 0,15 euros par 10 g **

Lettre service économique DOM-TOM :

poids	DOM	TOM
jusqu'à * 20 g	0,50 euros	0,50 euros
plus de 20 g	Tarif ME + 0,04 euros par 10 g **	Tarif ME + 0,10 euros par 10 g **

Renseignements complémentaires

\* poids jusqu'à 20 g signifie jusqu'à 20 g inclus

\*\* Tarif MR + 0,06 euros par 10 g pour une lettre service rapide DOM est à interpréter de la façon suivante.

Pour une lettre de 21 grammes c'est 0,73 euros ( 0,55 + 3 \* 0,06)

Pour une lettre de 30 grammes c'est 0,73 euros ( 0,55 + 3 \* 0,06)

Pour une lettre de 31 grammes c'est 0,79 euros ( 0,55 + 4 \* 0,06)

## 06-\*\*\*-Calcul d'une facture d'électricité

Consignes : ne pas utiliser les opérateurs logiques ET, OU, NON ; n'utiliser que les opérateurs relationnels <, >, <=, >=, =, <>

On désire calculer le montant d'une facture d'électricité sachant que l'abonné paye :

- des frais fixes d'abonnement de 25 €
- sa consommation selon un tarif à tranches
  - 0,20 € par kWh pour les 100 premiers kWh
  - 0,35 € par kWh pour les 150 suivants
  - 0,20 € par kWh pour ceux qui excèdent 250 kWh

On connaît pour l'abonné le relevé du compteur :

variable numérique : AI l'ancien index

variable numérique : NI le nouvel index

On désire la somme à payer dans la variable numérique nommée "montant"

Rédiger l'algorithme..

# CORRIGES

## 01-\* - Simulation d'instructions conditionnelles

```
algorithme test_1
```

```
/* Déclarations */
```

```
Variables
```

```
    heure          : entier
```

```
Début
```

```
    écrire ("Heure entre 0 et 23")
```

```
    lire (heure)
```

```
    si heure > 12 alors
```

```
        écrire ("soir")
```

```
    sinon
```

```
        écrire ("matin'")
```

```
    finsi
```

```
Fin
```

```
algorithme test_2
```

```
/* Déclarations */
```

```
Variables
```

```
    heure          : entier
```

```
Début
```

```
    écrire ("Heure entre 0 et 23")
```

```
    lire (heure)
```

```
    si heure >= 0 ET heure <= 23  
        alors
```

```
        si heure > 12 alors
```

```
            écrire ("soir")
```

```
        sinon
```

```
            écrire ("matin'")
```

```
        finsi
```

```
    finsi
```

```
Fin
```

### Solution

Il faut numéroter toutes les lignes. A gauche test\_1, à droite test\_2

simulation n°1

ligne	HEURE	écran	ligne	HEURE	écran
1		Heure entre 0 et 23	1		Heure entre 0 et 23
2	<b>10</b>		2	<b>10</b>	
3			3		
5		matin	4		
6			5		
			7		matin
			8		
			10		

simulation n°2

ligne	HEURE	écran	ligne	HEURE	écran
1		Heure entre 0 et 23	1		Heure entre 0 et 23
2	<b>15</b>		2	<b>15</b>	
3			3		
4		soir	4		
6			5		
			6		soir
			8		
			10		

simulation n°3

ligne	HEURE	écran	ligne	HEURE	écran
1		Heure entre 0 et 23	1		Heure entre 0 et 23
2	<b>25</b>		2	<b>25</b>	
3			3		
4		soir	9		erreur
6			10		

## 02-\* -Conditionnelles et séquence

**algorithme** test\_1

*/\* Déclarations \*/*

**Variables**

nb : entier

**Début**

```
1 écrire("Donner un nombre
entier")
2 lire (nb)
3 si nb <= 0 alors
4   nb ← nb + 5
   sinon
5   nb ← nb - 5
6 finsi
7 écrire("maintenant le nombre
vaut : " , nb)
```

**Fin**

**algorithme** test\_2

*/\* Déclarations \*/*

**Variables**

heure : entier

**Début**

```
1 écrire("Donner un nombre
entier")
2 lire (nb)
3 si nb <= 0 alors
4   nb ← nb + 5
5 finsi
6 si nb > 0 alors
7   nb ← nb - 5
8 finsi
9 écrire ("maintenant le nombre
vaut : " , nb)
```

**Fin**

### Solution

algo 1

	NB	$NB \leq 0$	écran
1			Donner un nombre entier
2	5		
3		<i>faux</i>	
4			
5	0		
6			
7			maintenant le nombre vaut : 0

algo 1

	NB	$NB \leq 0$	écran
1			Donner un nombre entier
2	0		
3		<i>vrai</i>	
4	5		
6			
7			maintenant le nombre vaut : 5

algo 1

	NB	$NB \leq 0$	écran
1			Donner un nombre entier
2	-5		
3		<i>vrai</i>	
4	0		
6			
7			maintenant le nombre vaut : 0

algo 2

avec 5

	NB	$NB \leq 0$	$NB > 0$	écran
1				Donner un nombre entier
2	5			
3		<i>faux</i>		
4				
5				
6			<i>vrai</i>	
7	0			
8				
9				maintenant le nombre vaut : 0

algo 2

avec 0

	NB	$NB \leq 0$	$NB > 0$	écran
1				Donner un nombre entier
2	0			
3		<i>vrai</i>		
4	5			
5				
6			<i>vrai</i>	
7	0			
8				
9				maintenant le nombre vaut : 0

algo 2

avec -5

	NB	$NB \leq 0$	$NB > 0$	écran
1				Donner un nombre entier
2	-5			
3		<i>vrai</i>		
4	0			
5				
6		<i>faux</i>		
8				
9				maintenant le nombre vaut : 0

**03-\*\*-Prix d'une place de cinéma**

Algorithme cinéma\_1

*/\* Déclarations \*/*

Variables

jour : chaîne de caractères */\* nom du jour {lu, ma, me, je, ve, sa, di} \*/*  
age : entier */\* age \*/*  
etud : caractère */\* indicateur du statut d'étudiant o pour étudiant \*/*  
prix : réel */\* prix de la place \*/*

*/\* Instructions \*/*

Début

lire(jour)

lire(age)

lire(etud)

si (jour = "me") ou (etud = 'o') ou ( (age < 18 ou age >= 65) et (jour <> "sa") et (jour <> "di") ) alors

    prix ← 4,5

sinon

    prix ← 7

finsi

Fin

Algorithme cinéma\_2

*/\* Déclarations \*/*

*/\* identiques \*/*

Début

lire(jour)

lire(age)

lire(etud)

si etud = 'o' alors

    prix ← 4,5

sinon

    si jour = "me" alors

        prix ← 4,5

    sinon

        si (jour = "sa" ) ou ( jour = "di" ) alors

            prix ← 7

        sinon

            si (age < 18) ou (age >= 65) alors

                prix ← 4,5

            sinon

                prix ← 7

            finsi

        finsi

    finsi

finsi

Fin



Algorithme cinéma\_3

/\* Déclarations \*/

/\* identiques \*/

Début

lire(jour)

lire(age)

lire(etud)

si etud = 'o' alors

    prix ← 4,5

sinon si jour = "me" alors

    prix ← 4,5

sinon si (jour = "sa" ) ou ( jour = 'di" ) alors

    prix ← 7

sinon si (age < 18) ou (age >= 65) alors

    prix ← 4,5

sinon

    prix ← 7

finsi

Fin

Algorithme cinéma\_4

/\* Déclarations \*/

/\* identiques \*/

Début

lire(jour)

lire(age)

lire(etud)

si jour = "me" alors

    prix ← 4,5

sinon

    si (jour = "sa") ou (jour = "di") alors

        si ETUD = 'o' alors

            prix ← 4,5

        sinon

            prix ← 7

        finsi

    sinon /\* autres jours \*/

        si (age < 18) ou (age >= 65) ou (etud = 'o') alors

            prix ← 4,5

        sinon

            prix ← 7

        finsi

    finsi

finsi

Fin

**04-\*\*-Calcul d'âge**

On suppose (non contrôlé) que la date de naissance est forcément antérieure à la date du jour

algorithme Age\_1

*/\* Déclarations \*/*

Variables

j1, m1, a1 : entier */\* date de naissance \*/*  
 j2, m2, a2 : entier */\* date du jour considéré \*/*  
 age : entier */\* âge calculé \*/*

*/\* Instructions \*/*

Début

```
lire(j1, m1, a1)
lire(j2, m2, a2)
si a1 = a2 alors
  age ← 0
sinon /* a2 > a1 */
  si m2 < m1 alors
    age ← a2 - a1 - 1
  sinon
    si m2 > m1 alors
      age ← a2 - a1
    sinon /* m1 = m2 */
      si j2 < j1 alors
        age ← a2 - a1 - 1
      sinon /* j2 ≥ j1 */
        age ← a2 - a1
        si j1 = j2 alors
          écrire ("bon anniversaire")
        finsi
      finsi
    finsi
  finsi
fini
fini
fini
fini
```

Fin

algorithme age\_2

*/\* Déclarations \*/*

*/\* identiques \*/*

Début

```
age ← a2 - a1
si (m2 < m1) ou (m2 = m1 et j2 < j1) alors
  age ← age - 1
fini
si j1 = j2 et m1 = m2 alors
  écrire ("bon anniversaire")
fini
```

Fin

**05-\*\*-Affranchissement**

Algorithme Affranchissement

*/\* Lit la masse et la destination d'une lettre de moins de 100 grammes, calcule puis affiche la valeur d'affranchissement en service économique et en service rapide*

*On suppose que l'utilisateur donne les valeurs attendues \*/*

*/\* Déclarations \*/*

Constantes

```
TARIF_RAP_20_FR      = 0.55
TARIF_RAP_50_FR      = 0.88
TARIF_RAP_100_FR     = 1.33
TARIF_ECO_20_FR      = 0.50
TARIF_ECO_50_FR      = 0.72
TARIF_ECO_100_FR     = 0.87
TARIF_RAP_20_DOM     = 0.55
TARIF_RAP_PLUS20_DOM = 0.06
TARIF_RAP_PLUS20_TOM = 0.15
TARIF_ECO_20_DOM     = 0.50
TARIF_ECO_PLUS20_DOM = 0.04
TARIF_ECO_PLUS20_TOM = 0.10
```

Variables

```
masse      : réel      /* masse de la lettre */
code       : caractère /* code destination {'m', 'd', 't' } */
prixEco    : réel      /* prix économique */
prixRap    : réel      /* prix rapide */
nbTranche  : réel      /* nb de tranches de 10 g */
```

Début

```
écrire ("masse entre 0 et 100 : ")
lire(masse)
écrire ("code de la destination : m, d ou t : ")
lire(code)
si masse ≤ 20 alors          /* identique pour métropole, DOM, TOM */
    prixEco ← TARIF_ECO_20_FR
    prixRap ← TARIF_RAP_20_FR
sinon
    si masse ≤ 50 alors      /* 20 < masse ≤ 50 */
        prixEco ← TARIF_ECO_50_FR
        prixRap ← TARIF_RAP_50_FR
    sinon                    /* 50 < masse ≤ 100 */
        prixEco ← TARIF_ECO_100_FR
        prixRap ← TARIF_RAP_100_FR
    finsi
finsi                        /* prixEco et prixRap ont la valeur de base */
si masse > 20 et DEST ≠ 'm' alors /* hors métropole */
    nbTranche ← masse div 10
    si masse mod 10 ≠ 0 alors     /* tranche supérieure */
        nbTranche ← nbTranche + 1
    finsi
    si DEST = 'd' alors          /* DOM */
        prixEco ← prixEco + TARIF_ECO_PLUS20_DOM * nbTranche
        prixRap ← prixRap + TARIF_RAP_PLUS20_DOM * nbTranche
    sinon                        /* TOM */
        prixEco ← prixEco + TARIF_RAP_PLUS20_TOM * nbTranche
        prixRap ← prixRap + TARIF_RAP_PLUS20_TOM * nbTranche
    finsi
finsi
écrire(prixEco, prixRap)
```

Fin

**06-\*\*\*-Calcul d'une facture d'électricité**

Algorithme Facture

/\* Déclarations \*/

Constantes

```
FRAIS_FIXES           = 25
TARIF_TRANCHE1        = 0.20
TARIF_TRANCHE2        = 0.35
TARIF_TRANCHE3        = 0.20
TAILLE_TRANCHE1       = 100
TAILLE_TRANCHE2       = 150
```

Variables

```
ai                    : réel      /* ancien index */
ni                    : réel      /* nouvel index */
conso                 : réel      /* consommation */
facture               : réel      /* montant de la facture */
```

/\* Instructions \*/

Début

```
écrire("Saisir l'ancien index : ")
lire(ai)
écrire("Saisir le nouvel indice : ")
lire(ni)
conso ← ni - ai
si (conso <= TAILLE_TRANCHE1) alors
  facture ← conso * TARIF_TRANCHE1
sinon
  si (conso <= TAILLE_TRANCHE2) alors
    facture ← TAILLE_TRANCHE1 * TARIF_TRANCHE1 +
              (conso - TAILLE_TRANCHE1) * TARIF_TRANCHE2
  sinon
    facture ← TAILLE_TRANCHE1 * TARIF_TRANCHE1 +
              TAILLE_TRANCHE2 * TARIF_TRANCHE2 +
              (conso - TAILLE_TRANCHE1 - TAILLE_TRANCHE2) *
              TARIF_TRANCHE3
  fsi
fsi
facture ← facture + FRAIS_FIXES
écrire ("Le montant de votre facture est : ", facture, " euros")
```

Fin