

TD N°1

Bases de l'algorithmique

1. Conversion de base

Complétez le tableau ci-dessous en inscrivant les valeurs des nombres décimaux exprimées dans les bases demandées :

Base 10 Décimal	Base 16 Hexadécimal	Base 8 Octal	Base 2 Binaire
8	0x08	10	10 00
12	0x0C	14	11 00
25	0x19	31	01 10 01
32	0x20	40	10 00 00
64	0x40	100	01 00 00 00
89	0x59	131	01 01 10 01
256	0x100	400	01 00 00 00 00

2. Affectation de variables

Complétez le tableau ci-dessous en inscrivant les nouvelles valeurs des 4 variables VarA, VarB, VarC et VarD après exécution des instructions de la première colonne.

Considérez qu'un espace mémoire dans lequel on n'a affecté aucune valeur ne contient pas de valeur pertinente.

Repérez les notations incorrectes et n'en tenez pas compte pour la suite du tableau

Instructions	Variables			
	VarA	VarB	VarC	VarD
VarB ← 4	-	4	-	-
VarC ← VarB + 10	-	4	14	-
VarA ← 0	0	4	14	-
VarD ← VarD + VarA	0	4	14	-
VarD ← VarA	0	4	14	0
VarB ← VarB * VarD	0	0	14	0
VarC ← VarB + 5	0	0	5	0
VarA ← 10 + 4 + 0	14	0	5	0
5 + VarB ← VarD				
VarC ← VarA + VarB + VarD	14	0	14	0

3. Calcul de la TVA

Ecrivez un programme qui acquiert un nombre décimal (« à virgule »), censé représenter un prix hors taxes, et qui affiche le prix TTC correspondant. Le taux de la TVA est de 19,6 %. Vous utiliserez les fonctions de lecture et d’affichage suivantes :

```
Procédure Lire(S x : Réel)
Procédure Ecrire(E x : Réel)
```

Corrections :

```
Programme calculPrixTTC()
Déclaration
    prixBrut : Réel
    TVA : Réel
    prixNet : Réel
Début
    Lire(prixBrut)
    TVA ← 19.6
    prixNet ← (1.0 + TVA / 100.0) * prixBrut
    Ecrire(prixNet)
Fin
```

4. Parité d’un nombre

Ecrivez une fonction booléenne, `estPair`, qui à partir d’un nombre entier strictement positif retourne VRAI si ce nombre est pair et FAUX sinon.

Corrections :

```
Fonction estPair(E x : Entier) : Booléen
Déclaration
Début
    Si x%2 = 0 alors
        Retourner(VRAI)
    Sinon
        Retourner(FAUX)
    FinSi
Fin
```

5. Machine qui rend la monnaie

Ecrivez un programme qui, pour une somme donnée en euros, sans centimes, imprime le nombre (minimal) de billets nécessaires pour la composer. Les billets disponibles sont respectivement de 100, 50, 20, 10 et 5 euros.

Exemple :

$$349 = 3 \times 100 + 0 \times 50 + 2 \times 20 + 1 \times 5 + 4 \text{ euros}$$

(Indication : a et b étant des expressions entières, le quotient et le reste de la division entière de a par b s’obtiennent respectivement par les expressions a / b et $a \% b$).

```
Procédure rendreMonnaie (E somme : Entier)
```

```
Déclaration
```

```
    nbBillets100 : Entier  
    nbBillets50  : Entier  
    nbBillets20  : Entier  
    nbBillets10  : Entier  
    nbBillets5   : Entier  
    reste        : Entier
```

```
Début
```

```
    nbBillets100 ← somme / 100  
    Ecrire(NbBillets100)  
    somme ← somme % 100  
    nbBillets50 ← somme / 50  
    Ecrire(NbBillets50)  
    somme ← somme % 50  
    nbBillets20 ← somme / 20  
    Ecrire(NbBillets20)  
    somme ← somme % 20  
    nbBillets10 ← somme / 10  
    Ecrire(NbBillets10)  
    somme ← somme % 10  
    nbBillets5 ← somme / 5  
    Ecrire(NbBillets5)  
    reste ← somme % 5  
    Ecrire(reste)
```

```
Fin
```