

S1-TD N°: 03

Exercice #1 (Homework):

- 1) Écrire un algorithme permettant d'échanger les valeurs de **deux** variables **A** et **B**, de même type.
- 2) Une variante du précédent : on dispose de **trois** variables **A**, **B** et **C**. Écrivez un algorithme transférant à B la valeur de A, à C la valeur de B et à A la valeur de C.

Exercice #2:

- 1) Que font les algorithmes suivants ?
- 2) Imaginer un problème auquel ils permettent de répondre.

```
Algorithmme Algo1;
Variables
|   N:entier;
|   montant:réel;
Début
|   Lire(N);
|   Si (N<=20) Alors
|   |   montant ← 0.20 * N;
|   FinSi;
|
|   Si (N>20) Alors
|   |   montant ← 0.20 * 20 + (N-20) * 0.10;
|   FinSi;
|
|   Ecrire(montant);
Fin.
```

```
Algorithmme Algo2;
Variables
|   N:entier;
|   montant:réel;
Début
|   Lire(N);
|   Si (N<=20) Alors
|   |   montant ← 0.20 * N;
|   Sinon
|   |   montant ← 0.20 * 20 + (N-20) * 0.10;
|   FinSi;
|
|   Ecrire(montant);
Fin.
```

Exercice #3:

Un cinéma propose les tarifs suivants pour les groupes :

8 DA la place pour les 5 premières ;

6 DA la place pour les suivantes jusqu'à 10 ;

5,50 DA la place, celles au-delà de 10.

- 1) Le responsable d'une association vient acheter des billets ; combien devra-t-il payer pour 4 places ? pour 9 places ? pour 15 places ?
- 2) Écrire un algorithme permettant d'obtenir le montant à payer lorsque le nombre de places est donné.
- 3) Le tester pour les valeurs de la question 1.

Exercice #4:

- 1) Écrire un algorithme / un programme en C qui demande un nombre de départ, et qui ensuite affiche les dix nombres suivants. Par exemple, si l'utilisateur entre le nombre 17, le programme affichera les nombres de 18 à 27.
- 2) Utiliser d'autre forme de boucle.

Exercice #5:

Écrire un algorithme / un programme en C qui demande un nombre compris entre 10 et 20, jusqu'à ce que la réponse convienne. En cas de réponse supérieure à 20, on fera apparaître un message : « Plus petit ! », et inversement, « Plus grand ! » si le nombre est inférieur à 10.