

TD Algo

1 Préambule

Écrivez un algorithme qui demande à l'utilisateur de saisir un nombre n puis affiche les n premiers nombres.

```
Programme Affiche_n_entiers
Variable
    Entier n, i
Debut
    Afficher('Saisir un nombre')
    Saisir(@n)
    Pour i <- 1 A n faire
        Afficher(n)
    Fin tant que
Fin programme Affiche_n_entiers
```

2 Exercice 0

Écrivez un algorithme qui demande à l'utilisateur de saisir un nombre n puis affiche les n premiers nombres impairs.

```
Programme Affiche_n_impairs
Variables
    Entier n, i
Debut
    Afficher('Saisir un nombre')
    Saisir(@n)
    Pour i <- 1 a n faire
        Afficher(2 * i - 1)
    Fin tant que
Fin programme Affiche_n_impairs
```

3 Exercice 1

Écrivez un algorithme qui demande à l'utilisateur de saisir un nombre n puis calcule la somme des n premiers entiers pairs en commençant par 2.

```
Programme Calcule_n_pairs
Variables
    Entier n, i, pair_n
Debut
    Afficher('Saisir un nombre')
    Saisir(@n)
    Pour i <- 1 a n faire
        pair_n <- pair_n + 2 * i
    Fin tant que
    Afficher(pair_n)
Fin programme Calcule_n_pairs
```

4 Exercice 2

Écrivez un algorithme qui, à partir d'un montant à épargner et un taux d'intérêt annuel, calcule le montant augmenté des intérêts pour les n années à venir.

```
Programme Taux_d_interet
Variables
    Entier n, i, taux, resultat
Debut
    Afficher('Saisir un montant epargne')
    Saisir(@resultat)
    Afficher('Saisir un taux d'interet exprime en pourcentage')
    Saisir(@taux)
    Afficher('Saisir un nombre d'annees')
    Saisir(@n)
    Pour i <- 1 a n faire
        resultat <- resultat + (resultat * taux / 100)
    Fin tant que
    Afficher('Dans ' & n & ' annees, le montant epargne sera de ' & resultat)
Fin programme Calcule_n_paires
```

5 Exercice 3

Écrivez un algorithme qui détermine si une année n est bisextile.

```
Programme Est_bisextile
Variables
    Entier n
Debut
    Afficher('Saisir une annee')
    Saisir(@n)
    Si ((n % 400 = 0) ou (n % 4 = 0) et (n % 100 <> 0)) alors
        Afficher(n & ' est bisextile')
    Sinon
        Afficher(n & ' n'est pas bisextile')
    Fin si
Fin programme Est_bisextile
```

6 Exercice 4

Écrivez un algorithme qui calcule l'impôt sur le bénéfice d'une société, le montant du bénéfice étant demandé à l'utilisateur.

```
Programme Impot
Variables
    Entier n
Debut
    Afficher('Saisir le montant du benefice')
    Saisir(@n)
    Si (n <= 10 000) alors
        Afficher('L'impot est de ' & n * 20 / 100)
    Sinon
        Si (n <= 15 000) alors
            Afficher('L'impot est de ' & 2000 + n * 26 / 100)
        Sinon
            Afficher('L'impot est de ' & 3300 + n * 22 / 100)
        Fin si
    Fin si
Fin programme Impot
```

7 Exercice 5

L'utilisateur devine un nombre pair compris entre 10 et 100 généré par l'ordinateur avec la fonction :
nombreAleatoire(entier n) : entier

```
Programme Fait_deviner_pair
Variables
    Entiers n, saisie, cpt
Debut
    n <- 2*(5 + nombreAleatoire(45))
    cpt <- 1
    Faire
        Afficher('Saisir nombre')
        Saisir(@saisie)
        Si (n > saisie) alors
            Afficher ('C'est plus')
            cpt++
        Fin si
        Si (n < saisie) alors
            Afficher ('C'est moins')
            cpt++
        Fin si
    Tant que (n <> saisie)
        Afficher('Bien joué, trouve en ' & cpt & ' coups')
Fin programme Fait_deviner_pair
```

8 Exercice 6

Écrire un programme qui devine un nombre pair (version dychotomique)

```
Programme Devine_pair
Variables
    Entiers proposition, min, max, i, n, cpt
    Caractere reponse
Debut
    min <- 0
    max <- 100
    cpt <- 0
    Afficher ("Choisissez un nombre de 0 a 100")
    Faire
        cpt <- cpt + 1
        proposition <- (max + min)/2
        proposition <- proposition + proposition % 2
        //Pairisation de proposition : si proposition
        //est impair, proposition % 2 vaut 1,
        //donc proposition devient pair
        Afficher("Le nombre est " & proposition)
        Saisir(@reponse)
        Si reponse = '+'
            min <- proposition + 2
        Fin si
        Si reponse = '-'
            max <- proposition - 2
        Fin si
    Tant que (reponse <> '=' )
        Afficher ("Trouve en " & cpt & " coups")
Fin programme Devine_pair
```

9 Cours : application 1

Écrire un programme qui échange le contenu de deux variables.

```
Programme Swap
Variables
    Entiers x, y, tmp
Debut
    //Init de x et y
    tmp <- x
    x <- y
    y <- tmp
Fin programme Swap
```

10 Cours : application 3

Écrire un programme cherchant dans un tableau le nombre 20

```
Programme Application_3
Constantes
    Entier taille = 20
Variables
    Tableau d'entiers tab[taille]
    Entier i
Debut
    //Saisie de tab
    //Affichage
    Tant que (i <= taille et tab[i] <> 20) faire
        i++
    Fin tant que
    Si ( i <= taille) alors
        Afficher ('Trouve')
    Fin si
Fin programme Application_3
```

11 Cours : application 4

Écrire un programme qui remplit un tableau à 2 dimensions de 10×10

```
Programme Table_de_multiplication
Variables
    Entiers i, j
    Tableau d'entiers tab[10,10]
Debut
    Pour i <- 1 a 10 faire
        Pour j <- 1 a 10 faire
            tab[i, j] <- i * j
        Fin pour
    Fin pour
Fin programme Table_de_multiplication
```

12 Bonus

1. Écrire un algorithme permettant à l'utilisateur de :
 - (a) Saisir et mémoriser n nombres compris entre 1 et 50
 - (b) Saisir et mémoriser n nombres distincts
 2. En déduire un algo permettant à un joueur de saisir une grille de loto où sont cochées 6 cases parmi 50.
1. (a)

```
Programme Saisie_n_nombres
Variables
    Entier i
    Tableau d'entiers tab[n]
Debut
    Pour i <- 1 a n faire
        Faire
            //Afficher...
            Saisir(@tab[i])
            Tant que (tab[i] < 1 ou tab[i] > 50)
        Fin pour
Fin programme Saisie_n_nombres
```

(b)

```
Programme Saisie_n_nombres_distincts
Variables
    Entier i
    Tableau d'entiers tab[n]
Debut
    Pour i <- 1 a n faire
        Faire
            //Afficher...
            Saisir(@tab[i])
            Tant que (tab[i] < 1 ou tab[i] > 50)
        Fin pour
Fin programme Saisie_n_nombres
```

13 Cours : application 5

```
Programme Nom_random
Type
    Enregistrement Stagiaire
        Chaîne Nom
        Chaîne Prenom
        Entier Age
    Fin enregistrement Stagiaire
Variables
    Entier i
    Stagiaire[20] stagiaires
Debut
    Pour i <- 1 a 20 faire
        //Afficher...
        Saisir(@stagiaires[i].Nom)
        //Afficher...
        Saisir(@stagiaires[i].Prenom)
        //Afficher...
        Saisir(@stagiaires[i].Age)
    Fin pour
Fin programme
```