

Programmation - Ada

Quelques exercices – Niveau 1 - Débutants

[Exercice 01 : Saisie et comparaison \(1\)](#)

[Exercice 02 : Saisie et comparaison \(2\)](#)

[Exercice 03 : Test d'une boucle for \(1\)](#)

[Exercice 04 : Test d'une boucle for \(2\)](#)

[Exercice 05 : Test d'une boucle for \(3\)](#)

[Exercice 06 : Test d'une boucle for \(4\)](#)

[Exercice 07 : Test d'une Boucle Tant Que \(1\)](#)

=====

Programmation - Ada

Quelques exercices – Niveau 1 - Débutants

Cette proposition de quelques exercices est un document provisoire pour répondre à une attente.

La présentation et l'écriture des programmes doit être conforme aux exemples données en cours, c'est-à-dire :

- La documentation au niveau du programme source,
- Des messages au niveau de l'exécution du programme.

Exercice 01 : Saisie et comparaison (1).

Nom du programme : Ada_Ex01

Sujet : Saisir un nombre entier N (entre 0 et 20 sans contrôle). Si N est strictement inférieur à 10 écrire le message "Vous n'avez pas le moyenne" dans le cas contraire écrire le message "Vous avez obtenu votre U.E."

Démarche et pseudo-code :

Donner la structure de ce programme (structure de bloc, structure alternative, structure répétitive, ...).

Ecrire le pseudo code.

Codification en Ada et commentaires :

Codifier ce programme en Ada.

Le squelette est le suivant :

```
-- *****
-- Nom Prog      : Ada_Ex01.adb
-- Type         : Application
-- Sujet        : Saisie et comparaison
--
-- Auteur       : R. VANDAELE
-- Version      : 1.0
-- Creation     : 10/10/2006
-- Dern. Modif : 10/10/2006
--
-- Compilateur  : GNAT 3.12p
-- Remarques    : Environnement Windows XP SP2
-- *****

-- Paquetage(s) utilise(s)

with Ada.Text_IO;
use  Ada.Text_IO;
```

```

procedure Ada-Ex01 is
  -- partie declarative
  -- Variables, Procedure, Fonctions, ...

begin
  -- Debut du corps de la procedure

  -- Message de debut d'execution
  Put_Line ("Debut - Execution Ada_Ex01");
  New_Line;
  Put_Line ("Saisie et Comparaison (1)");
  Put_Line ("-----");
  New_Line;

  -- Procedure proprement dite
  -- Suite d'instructions

  -- Message de fin d'execution
  New_Line;
  Put_Line ("Fin   - Execution Ada_Ex01");

end Ada_Ex01;

```

Donner le résultat d'une exécution :

Combien faut-il d'exécution(s) pour prouver que votre programme fonctionne correctement ?

Exercice 02 : Saisie et comparaison (2).

Nom du programme : Ada_Ex02

Sujet : Saisir un nombre entier N (entre -20 et +40 sans contrôle).

Si N est strictement inférieur à 0 écrire le message "Il fait très froid et il gèle"

Si N est égal à 0 écrire le message "L'eau se transforme en glace"

Si N est strictement supérieur à 10 écrire le message "La température est positive"

Si N est strictement supérieur à 20 écrire le message "Il commence à faire bon"

Démarche et pseudo-code :

Donner le structure de ce programme (structure de bloc, structure alternative, structure répétitive, ...).

Ecrire le pseudo code.

Codification en Ada :

Codifier ce programme en Ada.

Donner le résultat d'une exécution :

Combien faut-il d'exécution(s) pour prouver que votre programme fonctionne correctement ?

Exercice 03 : Test d'une Boucle For (1)

Nom du programme : Ada_Ex03

Sujet : Tester la boucle for suivante :

```
For I in 1 .. 15 loop
    Put ('*');
end loop;
```

Codification en Ada :

Codifier ce programme complet en Ada.

Donner le résultat d'une exécution :

Expliquer le fonctionnement de ce programme.

Exercice 04 : Test d'une Boucle For (2)

Nom du programme : Ada_Ex04

Sujet : Tester la boucle for suivante :

```
For I in reverse 5 .. 15 loop
    Put (I, width => 4);
end loop;
```

Codification en Ada :

Codifier ce programme complet en Ada.

Donner le résultat d'une exécution :

Expliquer le fonctionnement de ce programme.

Exercice 05 : Test d'une Boucle For (3)

Nom du programme : Ada_Ex05

Sujet : Tester la boucle for suivante :

```
For I in 1 .. 3 loop
  For J in 1 .. I loop
    Put ('*');
  end loop;
  new_line;
end loop;
```

Codification en Ada :

Codifier ce programme complet en Ada.

Donner le résultat d'une exécution :

Expliquer le fonctionnement de ce programme.

Exercice 06 : Test d'une Boucle For (4)

Nom du programme : Ada_Ex06

Sujet : Tester la boucle for suivante :

```
For I in 1 .. 2 loop
  For J in 1 .. 3 loop
    Put (" I = ");
    Put (I, width => 4);
    Put (" J = ");
    Put (J, width => 4);
    New_line;
  end loop;
end loop;
```

Codification en Ada :

Codifier ce programme complet en Ada.

Donner le résultat d'une exécution :

Expliquer le fonctionnement de ce programme.

Exercice 07 : Test d'une Boucle Tant Que (1)

Nom du programme : Ada_Ex07

Sujet : Saisir une borne de début et une borne de fin. Donner tous les nombres avec un pas de 3 depuis la borne de début jusqu'à la borne fin

Exemple : Début 11 Fin 28 Visualisation : 11 14 17 20 23 26

Démarche et pseudo-code :

Donner le structure de ce programme (structure de bloc, structure alternative, structure répétitive, ...).

Ecrire le pseudo code (uniquement la partie du Tant Que).

Codification en Ada :

Codifier ce programme complet en Ada.

Donner le résultat d'une exécution :

Commenter le ou les résultats si nécessaire.

=====

Quand vous avez terminé ces programmes, envoyez-moi un message et je vous donnerai la suite.

=====