

REINET : ADA

Mr. Brian FRAVAL

REINET : ADA

par Mr. Brian FRAVAL

<http://education.fraval.org/>

Copyright (c) 2002 Brian FRAVAL (<mailto:brian@fraval.org>)

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, with the no Front-Cover Texts, and with no Back-Cover Texts.

A copy of the license is included in this site : <http://www.gnu.org/copyleft/fdl.html>

Historique des versions

Version version 0.1 29 novembre 2002

Version initiale

Table des matières

1. ADA	1
Installation de GNAT	1
Voici la liste des instructions ADA indispensables.....	1
Notion de type	??
Le type Integer.....	??
Le type Boolean.....	??
Le type énumératif.....	??
Le type Character	??
Le type Float.....	??
Le type tableau	??
Notion de procédure.....	??
Notion de fonction	??
Package/Module.....	??
Les algos récursifs.....	??
Les exceptions.....	??
2. Téléchargement du cours	??
Version PDF	??
Version RTF	??
Version PS	??
Bibliographie	??

Chapitre 1. ADA

Le langage de programmation ADA.

Installation de GNAT

L'installation du compilateur GNAT n'est pas très compliqué. Par exemple, voici la ligne de commande qu'il faut utiliser pour installer le compilateur sur la distribution Debian :

```
apt-get install gnat
```

Si vous avez une autre distribution, ou si vous utilisez un autre système que Linux, je vous conseil de lire cette documentation réalisée par Alexandre TOPOL du CNAM :

<http://deptinfo.cnam.fr/Enseignement/CycleProbatoire/Vari/Proj/GNAT/>
(<http://deptinfo.cnam.fr/Enseignement/CycleProbatoire/Vari/Proj/GNAT/>)

Voici la liste des instructions ADA indispensables

Instruction	Description
begin	Démarrer un programme
end	Terminer un programme
integer	Déclaration de variable de type entier
long_integer	Déclaration de variable de type entier long
float	Déclaration de variable de type réel. En ADA les nombres réels s'écrivent avec un point (24.5895).
long_float	Déclaration de variable de type réel long
character	Déclaration de variable contenant un seul caractère (lettre, chiffre ou symbole)
string (1..n)	Déclaration de variable de type chaîne de caractère.
constante	Déclaration de constante
subtype	Pour définir des sous-types
put("texte")	Affiche le mot "texte" à l'écran
put(nom_de_la_variable)	Affiche à l'écran le contenu de la variable "nom_de_la_variable"
put_line("texte")	Affiche le mot "texte" et passe à la ligne
new_line	Fait passer le curseur à la ligne
skip_line	Saute une ligne à l'écran et vide le tampon mémoire
get(nom_de_la_variable)	Enregistre la saisie au clavier de l'utilisateur dans la variable "nom_de_la_variable"
float(A)	Convertit une valeur entière A en une valeur réelle

Instruction	Description
integer(B)	Convertit une valeur réelle B en une valeur entière
A mod B	Donne le reste de la division de A par B
a**b	Calcule a puissance b
log(a)	Calcule le logarithme népérien de y

Notion de type

ADA est un langage très typé, il faut obligatoirement définir un type à tous les objets (variables, fonctions, procédures, etc.) du programme que vous réaliser.

Un type définit l'ensemble des valeurs que peut prendre un objet, ainsi que les opérations permises sur ses valeurs. En effet, certaines opérations sont possible que pour un type donné. Cela impose au développeur d'avoir une bonne connaissance des objets qu'il maîtrise.

Le type Integer

TODO

Les autres types entiers : subtype Entier is Integer range 1..200;

Le type Boolean

FALSE, TRUE

Le type énumératif

type Jour is (Lundi, Mardi, Mercredi, Jeudi, Vendredi, Samedi, Dimanche);

Le type Character

TODO

Le type Float

TODO

Le type tableau

```
type NumJour is Array(1..31) of Integer;
```

Notion de procédure

TODO

Notion de fonction

TODO

Package/Module

TODO

Les algos récursifs

TODO

Les exceptions

TODO

Chapitre 2. Téléchargement du cours

Version PDF

<http://education.fraval.org/fr/informatique/ada/ada.pdf>

Version RTF

<http://education.fraval.org/fr/informatique/ada/ada.rtf>

Version PS

<http://education.fraval.org/fr/informatique/ada/ada.ps>

Bibliographie

CANESI, Gérard et SUC, René. *Initiation et auto-formation à la programmation informatique*. ellipses,
Août 2001