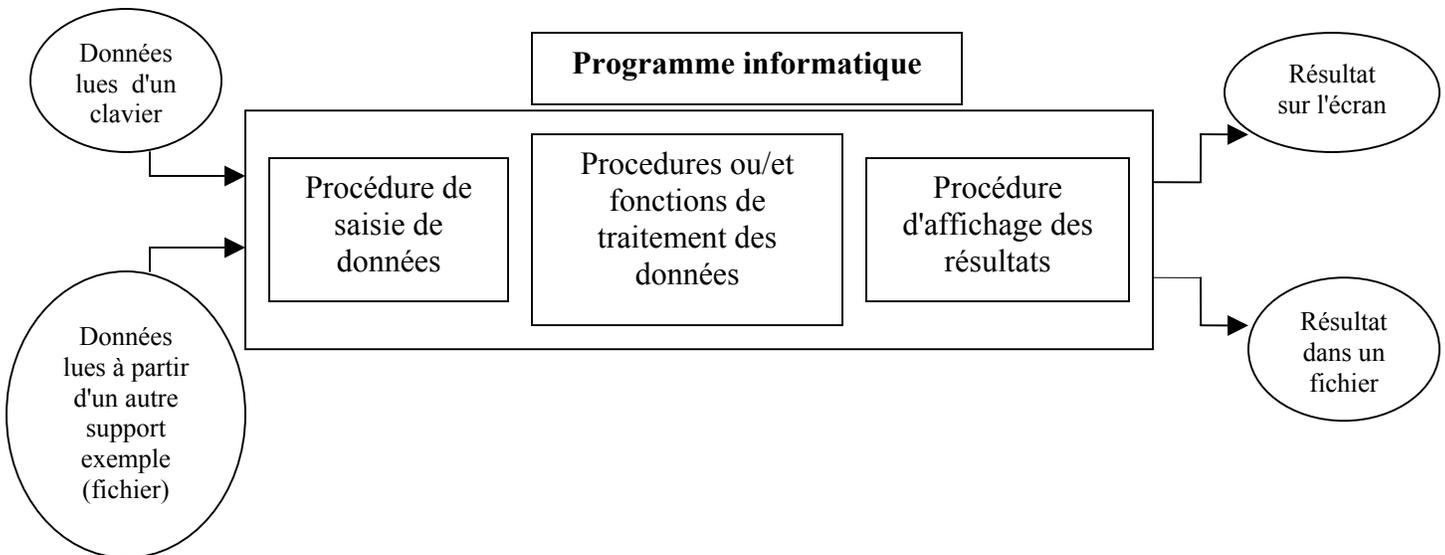


**Série d'exercices**  
**LES PROCEDURES ET FONCTIONS**

On reprend les exercices des séries 1 et 2, et de réécrire le code en utilisant des procédures et/ou des fonctions.

On a besoin d'une procédure pour la saisie des données, d'une procédure pour l'affichage des résultats et d'une ou plusieurs fonctions et/ou d'une ou plusieurs procédures pour les différents traitements demandés.



**Exercice** : Ecrire un programme **complet en ADA** qui demande les valeurs d'un **entier N**, et calcule la somme ci-dessous puis l'affiche.

Prévenir le dépassement du type Integer.  $Somme = 1^2 + 2^2 + \dots + N^2$  ;

---

```
-- Nom fichier      : Calculer_Somme.adb
-- Auteur           : Toto
-- Date             : 26/10/2008
-- But              : Calculer la somme des carrés d'un nombre entre au clavier par l'utilisateur
-- Dates de modif. : uniquement s'il y a des modifications
-- Raison           :
-- Modules appeles  : Text_IO, Integer_Text_IO
-- Mat. Particulier : matériel nécessaire pour l'utilisation de ce module.
-- Environnement   :
-- Compilation      : GNAT (ADA95)
-- Mode d'execution : Console
```

---

```

with Ada.Text_IO;          use Ada.Text_IO;
with Ada.Integer_Text_IO; use Ada.Integer_Text_IO;

procedure Calculer_Somme is

  -- Specification des procedures et fonctions
  function Carre(Terme : in Integer) return Integer ;
  procedure Somme_Carre(NbElements_Formel : in Integer; Somme_Formelle : out Integer);
  procedure Saisir(NbElements_Formel : in out Integer);
  procedure Afficher(NbElements_Formel : in Integer ; Somme_Formelle : in Integer) ;

  function Carre(Terme : in Integer) return Integer is
  begin
    return Terme * Terme;
  end Carre;

  procedure Somme_Carre(NbElements_Formel : in Integer ; Somme_Formelle : out Integer) is
    I : Integer;
  begin
    Somme_Formelle :=0;
    I :=0;
    while I < NbElements_Formel and Integer'Last- Somme_Formelle >I loop
      Somme_Formelle:= Somme_Formelle + Carre(I);
      I:= I+1;
    end loop;
  end Somme_Carre;

  procedure Saisir(NbElements_Formel : in out Integer) is
  begin
    Put(" Entrer La valeur : ");
    Get(NbElements_Formel);
    Skip_Line;
  end;

  procedure Afficher(NbElements_Formel : in Integer; Somme_Formelle : in Integer) is
  begin
    Put(" la somme des carres de 1 a" & Integer'Image(NbElements_Formel) & " est : ");
    Put(Somme_Formelle,0);
  end;

  -- declaration des variables
  N      : Integer;  -- la nombre N introduit par l'utilisateur
  Somme  : Integer;  -- la somme des carres

begin -- calcul_Somme
  Saisir(N);
  Somme_Carre(N, Somme);
  Afficher(N, Somme);
end Calculer_Somme;

```