Corrections exercices CM Ada

F. Singhoff

9 novembre 2015

1 Exercice 1

– Fichier `calcul.adb` :

```
package body calcul is

   function additionne(a : in integer;
      b : in integer)
        return integer is
   begin
      return a+b;
   end additionne;

   function multiplie(a : in integer;
      b : in integer)
        return integer is
   begin
      return a*b;
   end multiplie;

   function soustrait(a : in integer;
      b : in integer)
        return integer is
   begin
      return a-b;
   end soustrait;

   function divise(a : in integer;
      b : in integer)
        return integer is
   begin
      return a/b;
   end divise;

end calcul;
```

– Fichier `sol_exo1.adb` :

```
with text_io;
use text_io;
```

```
with calcul;

procedure sol_exo1 is

a : integer;
begin
    a:=calcul.additionne(calcul.multiplie(2,3),4);
    put_line(integer'image(a));
end sol_exo1;
```

2 Exercice 2

- ligne 1 : non (car types incompatibles)
- ligne 2 : non (car types incompatibles)
- ligne 3 : non (car types incompatibles)
- ligne 4 : oui
- ligne 5 : non (car dépassement de capacité)

3 Exercice 3

```
with text_io;
use text_io;

procedure exo3 is

i : integer :=0;
s : integer :=0;

package io is new text_io.integer_io(integer);
use io;

begin
        loop
                Put("Valeur = ");
                Get(i);
                s:=s+i;
                new_line;
                put("Somme = ");
                put(s);
                new_line;
        end loop;
exception
        when Data_Error => put_line("Erreur saisie");
end exo3;
```

4 Exercice 4

- page 57 : toutes activées sur le `begin` de la procédure principale. Aucune terminaison car `Ma_Tache` est dans une boucle. Ce programme contient 2 tâches.
- page 58 : toutes activées sur le `begin` de la procédure principale. Aucune terminaison car les tâches `T1, T2 et T` sont dans une boucle. Ce programme contient 13 tâches.

– page 59 : la procédure principale est activée sur le `begin` de la procédure principale. Les tâches `T` sont activées sur appel du `new`. Aucune terminaison car les tâches `T` sont dans une boucle. Ce programme contient 4 tâches.

5 Exercice 5

```
with Text_Io; use Text_Io;

procedure suite is

   package io is new text_io.integer_io(integer);
   use io;

   task suite1;
   task body suite1 is
   U0 : integer := 1;
   U1 : integer := 0;
   begin
      loop
        U1:=U0*2;
        Put("Un = "); Put(U1); new_line;
        delay 1.0;
        U0:=U1;
      end loop;
   end suite1;

   task suite2;
   task body suite2 ...
   --  La tache "suite2" implémente la suite Un=Un-1+2
   --  avec la même approche utilisée dans "suite1"
   end suite2;

begin
   null;
end suite;
```