Correction devoir Surveillé 1

22/03/2010, Licence 1 MASS semestre 2
Durée 0h45

Exercice 1

a- Un algorithme est un ensemble fini d'instructions correctement définis qui a pour but la résolution d'une tâche à partir d'un état initial et se terminant sur un état final bien défini.

b- Voir le cours cm01 !

Exercice 2

a-
```
/***************************************
* programme Processing qui dessine un Z
*
* entrée :
*   - aucune
*
* sortie :
*   - aucune (affichage à l'écran)
***************************************/
void setup() {
  fill(0);
  rect(30,20,50,10);
  for (int i=0; i<3 ; i++)
    rect(60-10*i,30+10*i,10,10);
  rect(30,60,50,10);
}
```

b- Ajouter l'instruction 'framerate(5); au débute de la méthode `setup()`.

Exercice 3

a- Notez qu'on ne demande pas de calculer le terme de la suite en finction de n mais en fonction des termes précédents U_{n+1} et U_n. Ce qui est strictement equivalent a calculer U_n en fonction de U_{n-1} et U_{n-2}. Ceci suppose donc que ces deux termes sont donnés, et qu'il suffit de les utiliser tels quels pour retourner la valeur du calcul :

```
/***************************************
* calcule le terme Un en fonction des
* termes precedents
*
* entrée :
*   - umoins1 : terme Un-1
*   - umoins2 : terme Un-2
*
* sortie :
```

```
*   - valeur de Un
****************************************/
float suiteU(float umoins1, float umoins2) {
  return 0.5*umoins1 + 2.0*umoins2;
}
```

b- Pour calculer la somme des Un il faut commencer par calculer chaque Un. Comme chaque Un depend des deux termes precedents, on a juste besoin de 3 variables: $a = U_n$, $b = U_{n-1}$ et $c = U_{n-2}$.

```
/****************************************
* calcul l'indice du premier terme qui approche la limite à 10^-2
*
* entrée :
*   - n
*
* sortie :
*   - Somme(ui), 0 <= i <= n
****************************************/
float sommeUi(int n) {
  if (n== 0) return 1;
  if (n== 1) return 1+2;

  float somme = 3;
  float c=1;
  float b=2;

  for (int i=2 ; i<=n ;i++) {
    float a = 0.5 *b + 2*c;
    somme = somme + a;
    c = b;
    b = a;
  }
  return somme;
}
```