TD1

October 23, 2024

1. Comment coder en `C` la fonction $sn(x) = 4.x.(\pi - x)/\pi^2$?

   Quelle fonction trigonométrique est approximée par le polynôme sn ?

2. Pour quelle valeur de v la fonction $sn(v) = 0.5$?

   Pour répondre à cette question, on utilisera d abord l algorithme de résolution du *point fixe* puis l algorithme de résolution *dichotomique*, et on précisera lequel est le plus rapide à fournir une solution.

3. Comment coder en `C` la fonction $cn(x) = 1 - \int_0^x sn(t).dt$?

   On rappel que, en considérant des valeurs *discrètes*, l expression précédente se ramène à la somme $\Sigma_{i=0}^{n} d.sn(i.d)$ avec $n = x/d$ et d une valeur arbitrairement petite.

   Quelle fonction trigonométrique est approximée par la fonction cn ?

4. On considère la suite récurrente définie par $x_0 = 0.123$ et $x_{n+1} = f(x_n)$ avec $f(x) = 3.99 \times x \times (1 - x)$.

   En affichant les 10 premières valeurs de x_n, dire en quoi la suite définie est “intéressante” ?

5. Comment définir le tableau cs tel que $cs[i] = sn(i.d)$ avec $i \in [0, 10]$ et $d = \pi/10$ et le tableau ps tel que $ps[i] = x_i$ avec $i \in [0, 2]$?

   Comment définir aussi le polynôme $p(i) = \Sigma_{j=0}^{2} ps[j].(i.d)^j$?

6. On pose l algorithme suivant:

   $erreur_0 = cs[0] - p(0)$

   $ps[i] = ps[i] + 0.1 \times erreur \times (i.d)^i$

   $erreur_1 = cs[0] - p(0)$

   Afficher les deux valeurs de *erreur* et discuter l intérêt de l’algorithme proposé.

7. Généraliser l algorithme précédent à toutes les valeurs de cs de façon à ce que l erreur globale soit infiniment petite.

   Comparer alors les valeurs de ps aux valeurs des coefficients du polynôme sn (de la question 1).