

TD n°3

Créer un sous-dossier TD_03 dans le dossier TD_PROG_IMP. Chaque exercice sera fait dans un sous-dossier appartenant au dossier TD_03. Plusieurs versions d'un même programme sont demandées. Il est alors impératif de conserver chacune des versions en dupliquant les fichiers sources. La compilation des programmes sera faite en ligne de commandes.

EXERCICE : Calcul d'un champ initial de concentration par une fonction porte

Dans le cadre du développement d'un programme de résolution de problèmes d'advection-diffusion 1D d'un scalaire passif (nous considérerons ici une concentration), une étape importante est l'initialisation de la distribution spatiale du scalaire. Dans le cas présent, l'utilisateur souhaite initialiser le champ de concentration par une distribution de type « porte » sur le domaine physique de longueur L et de direction x . Le champ initial de concentration est défini par :

$$C_{init}(x) = C_0 + (C_1 - C_0) \times (H(x - x_d) - H(x - x_f)) \quad (1)$$

où C_0 et C_1 sont les concentrations minimales et maximales dans le domaine, x_d et x_f sont les positions de début et de fin de la porte ($0 \leq x_d < x_f \leq L$) et $H(x)$ est la fonction d'Heaviside définie par :

$$H(x) = \begin{cases} 0 & \text{si } x < 0 \\ 1 & \text{si } x \geq 0 \end{cases} \quad (2)$$

Un exemple de champ initial de concentration est présenté sur la figure 1.

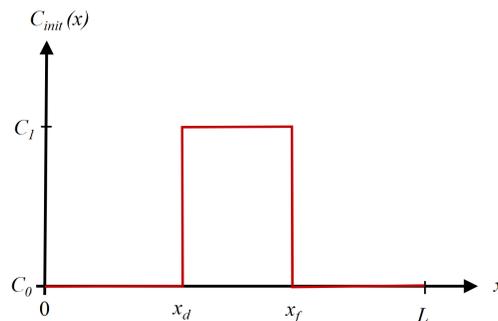


Figure 1 : Champ initial de concentration défini par une fonction « porte » (voir éq. 1).

1°) Créer la fonction de Heaviside (au sens du Fortran) appelée H , qui calcule la valeur $H(x)$ définie par (2) à partir d'une valeur quelconque d'un réel x .

2°) Ecrire le programme qui demande à l'utilisateur une valeur de x et affiche à l'écran le résultat de la fonction H .

3°) Créer la subroutine, notée *recup_donnees*, qui demande à l'utilisateur les valeurs de L , C_0 , C_1 , x_d et x_f .

4°) Ecrire le programme qui demande à l'utilisateur les valeurs de L , C_0 , C_1 , x_d et x_f .

5°) Créer la fonction, notée C_{init} , qui calcule la valeur de la fonction définie par (1) à partir d'une valeur quelconque d'un réel x et de celles de C_0 , C_1 , x_d et x_f (indication : il est possible d'utiliser une fonction pour faire un calcul dans une autre fonction).

6°) Ecrire le programme qui demande à l'utilisateur une valeur de L , C_0 , C_1 , x_d et x_f et affiche à l'écran le résultat de la fonction C_{init} pour $x=0$, $x=(x_d + x_f)/2$ et $x=L$ (indication : le programme devra faire appel à la subroutine *recup_donnees* ainsi qu'aux fonctions H et C_{init}).

Pour aller plus loin :

Le domaine physique de longueur L et de direction x est discrétisé en N points de calcul x_k régulièrement espacés et qui sont numérotés de $k=1$ à $k=N$, tels que $x_1 = 0$ et $x_N = L$. On définit deux vecteurs de 100 éléments réels, notés respectivement x_{reg} et C (correspondant à la position de chaque point du maillage régulier et à la concentration en ce même point du maillage).

7°) Modifier la subroutine *recup_donnees* pour qu'elle inclue la valeur de N dans la demande à l'utilisateur ($N \leq 100$).

8°) Ecrire le programme qui demande à l'utilisateur la valeur de N , L , C_0 , C_1 , x_d et x_f , calcule les tableaux x_{reg} et C et affiche leur valeur à l'écran, sous la forme :

Au point numéro $k = 1$, $x = 0$, $C = \dots$
 Au point numéro $k = 2$, $x = \dots$, $C = \dots$
 ...