

## I. Description :

Le débbugger est un outil permettant de suivre pas à pas l'execution d'un programme : connaissance à tout instant des valeurs des variables, position de *breakpoints* (points d'arrêt) pour stopper le programme à l'endroit voulu,...

**Avant toute chose vous devez avoir compilé vos fichiers objets (.o) avec l'option -g**

## II. Quelques commandes utiles :

### *Pour l'arrêt de l'execution*

#### **break**

- break FONCTION : pose un *breakpoint* à l'entrée de la FONCTION.
- break LIGNE : pose un *breakpoint* à la ligne LIGNE.
- break FICHIER:LIGNE : pose un *breakpoint* à la ligne LIGNE du FICHIER.
- break FICHIER:FONCTION : pose un *breakpoint* à l'entrée de la FONCTION du FICHIER.

#### **watch**

- watch EXPRESSION : le debugger s'arrete dès que l'EXPRESSION est modifiée.

#### **info**

- info : Affiche la liste de tous les *breakpoint* et *watchpoint*

#### **clear**

- clear FONCTION : efface les *breakpoint* de la FONCTION du fichier courant.
- clear FICHIER:FONCTION : efface les *breakpoint* de la FONCTION du fichier FICHIER.

### **delete**

- delete : efface les *breakpoint* ou les *watchpoint*

### **Pour le contrôle de l'exécution**

#### **run (r)**

- run ou r : démarre gdb

#### **step (s)**

- step COMPTE : fait continuer l'exécution du programme jusqu'à ce que COMPTE lignes de code soit passées (même si il y a des *breakpoint*)

#### **next (n)**

- next COMPTE : fait continuer l'exécution du programme jusqu'à ce que COMPTE lignes de code soit passées

#### **continue (c)**

- continue : continue l'exécution du programme

#### **finish**

- finish : arrête l'exécution juste après la fin de la fonction courante et affiche la valeur retournée par la fonction si celle-ci en renvoie une

### **Pour étudier les données et le code source**

#### **print (p)**

## Le débbuger GDB

Écrit par MACS 2009

Vendredi, 17 Juillet 2009 14:28 - Mis à jour Samedi, 30 Janvier 2010 18:35

---

- print EXPRESSION : affiche la valeur de EXPRESSION

### **display (disp)**

- display EXPRESSION : affiche la valeur de EXPRESSION

### **list (l)**

- list : affiche 10 lignes de code par défaut.
- list LIGNE : affiche les lignes centrées autour de la ligne LIGNE
- list FONCTION : affiche les lignes centrées autour du début de la FONCTION
- list : affiche plus de ligne
- list - : affiche les lignes précédents les dernière affichées

### **set**

- set VARIABLE=VALEUR : modification de la valeur d'une variable