

IDE Anjuta DevStudio (partie 1/2)

TP n°5 - Hoerd Mickaël/Quirin Arnaud - Avril 2004

Résumé

Anjuta est un projet OpenSource permettant de développer des applications complètes à l'aide d'un Environnement de Développement Intégré (ou IDE). Anjuta peut être téléchargé depuis le site <http://anjuta.sourceforge.net>. La version courante d'Anjuta au 13/04/2004 (1.2.2) contient un éditeur puissant (syntax highlighting, name completion, code hiding, ...), dispose d'un débogueur intégré, permet de gérer de multiples langages (C, C++, Java, Perl, Pascal, ...) et de développer des interfaces graphiques sous GTK, QT ou Gnome.

Note : Le TP utilise la version 0.1.9 d'Anjuta, installée sur les distas (ssh dista[1..20].ustrasbg.fr). Vous pouvez obtenir une aide en ligne en allant dans le menu "Help" de l'interface.

1 Création d'un projet

- Lancer Anjuta.
- Créer un nouveau projet.
- Demander les informations de votre projet (l'auteur, le nom du programme, le nombre de fichiers, ...).
- Rajouter le code de `/users/prof/quirina/GL/TP5/main.c` dans votre projet.
- Construire le projet (commande `build`) et le tester (commande `execute`).
- Formater et indenter le code source automatiquement.
- Tester la complétion des noms de fonction (taper une lettre puis `CTRL+ENTER`).
- Se débrouiller pour que le `main` prenne ses arguments en ligne de commande et demander à Anjuta de lancer le programme en spécifiant cette ligne de commande.
- Rajouter `#include "main.h"` en haut de votre fichier source. Paramétrer Anjuta pour que les fichiers `#include` soit pris dans le dossier `/users/prof/quirina/GL/TP5/`.
- Compiler et observer la ligne de commande exécutée par Anjuta pour la compilation.

2 Debugage

- Afficher le numéro des lignes, la toolbar "debug", et la colonne indiquant la position des breakpoints (marker margin).
- Lancer le débogueur.
- Placer un breakpoint sur la ligne `"z = factor(x-1);"`.
- Détacher le sous-menu "execution" du menu "debug" (cliquer sur "—————" en haut du menu).
- Dans la fenêtre des messages (`CTRL+F1`), afficher l'onglet "Local". Cette fenêtre montre le contenu des variables du contexte courant (définies dans un bloc `{...}`).
- En utilisant la toolbar, le menu détaché ou les touches de raccourci, exécuter le code pas à pas et noter l'évolution de la valeur des variables. L'onglet "Debug" de la fenêtre des messages indique le nom de la fonction courante, avec ses paramètres éventuels.

3 Debugage (2)

- Mettre un breakpoint sur la fonction `printf()` de la fonction `boucle()` et appeler cette fonction dans `main()`. Désactiver tout breakpoint dans la fonction `factor()`.
- Afficher l'évaluateur d'expressions (Watch Window).
- Lancer le débogueur et exécuter `boucle()` pas à pas.
- Ajouter les expressions dans l'évaluateur (`CTRL+P`) permettant d'afficher la valeur de `i` et de sa factorielle durant l'exécution de `boucle()`, et observer le résultat.

4 Divers

- Sauvegardez votre projet.
- Anjuta a créé un dossier `$HOME/Projects`. Explorez-le et recherchez vos sources.
- Editez quelques fichiers dont `Makefile.in` et `Makefile`.
- Suivre les instructions du fichier `INSTALL` pour débiter l'installation de votre programme.
- Importation d'un projet existant : créez un nouveau projet et importez les fichiers de l'un de vos projets personnels et le compiler.