

NPI - Calculatrice en Notation Polonaise Inverse

Auteur: MiKaël NAVARRO

Les notes qui suivent sont distribuées suivant les termes de la Licence de Documentation Libre GNU (GNU Free Documentation License, version 1.2 ¹) de la Free Software Foundation.

Cela signifie que vous pouvez copier, modifier et redistribuer ces notes tout à fait librement, pour autant que vous respectiez un certain nombre de règles qui sont précisées dans cette licence.

Pour l'essentiel, sachez que vous ne pouvez pas vous approprier ces notes pour les redistribuer ensuite (modifiées ou non) en définissant vous-même d'autres droits de copie. Les notes que vous redistribuez, modifiées ou non, doivent obligatoirement inclure intégralement le texte de la licence citée ci-dessus ainsi que le présent avis.

Vous êtes autorisé à demander une contribution financière à ceux à qui vous redistribuez ces notes, mais la somme demandée ne peut concerner que les frais de reproduction. Vous ne pouvez pas redistribuer ces notes en exigeant pour vous-même des droits d'auteur, ni limiter les droits de reproduction des copies que vous distribuez.

Ces notes sont publiées dans l'espoir qu'elles seront utiles, mais sans aucune garantie.

Contents

| | |
|---|----------|
| Présentation | 1 |
| Pourquoi ? | 1 |
| Installation | 2 |
| Configuration requise | 2 |
| Installation à partir des sources (tarball) | 2 |
| Mentions légales | 3 |
| Auteur | 3 |
| Licence | 3 |
| Distribution | 3 |
| Notes techniques | 3 |
| Algorithme | 3 |
| Implémentation | 4 |
| Outils | 4 |
| IRC bot | 4 |
| Jabber bot | 4 |
| Bibliographie | 4 |

Présentation

Pourquoi ?

Ce petit logiciel est une calculatrice, pour l'interpreteur Python.

Pourquoi, me direz-vous alors, implémenter une calculatrice alors qu'il en existe tant d'autres (bc, ...) ?

Et bien parcequ'elle est en Notation Polonaise Inverse, comme la HP48 !

Dans les années 20, Jan Lukasiewicz a développé un système de logique formelle permettant aux expressions mathématiques d'être indiquées sans parenthèses en plaçant les opérateurs avant (notation de préfixe) ou après (notation de suffixe) les opérandes.

La notation de préfixe est également connue en tant que notation polonaise en l'honneur de Lukasiewicz.

Et la notation de suffixe est nommé notation polonaise inversée (RPN = Reverse Polish Notation) également en l'honneur de Lukasiewicz.

Possédant une HP48GX, j'ai voulu avoir l'équivalent pour le shell Unix. Certe, le projet x48 existait déjà, mais mon idée était d'avoir un outil léger et rapide fonctionnant sur la plupart des plates-formes Unix.

Fonctionnalités :

- Portabilité ;
- Fonctions de manipulations de la pile ;
- Implémentations des fonctions mathématiques (module math) ;
- Ajout de fonctions de probabilités ;
- Complétion des commandes.

Installation

Configuration requise

Théoriquement ce logiciel doit fonctionner sur toutes les plates-formes où un interpréteur Python est disponible.

Testé avec succès sous :

- Debian Woody 3.1r0a (i686, Kernel 2.4.27) ;
- Cygwin (sur Win2000) ;
- Sunfire (Sparc 64, Sun Solaris 2.8) ;
- et SymbianOS (Nokia N95).

Installation à partir des sources (tarball)

Décompresser les sources avec

```
$ gzip -dc npi-x.x.x.tar.gz | tar xvf -
```

Pour lancer npi il suffit alors de taper

```
$ cd npi-x.x.x
$ python -O ./npi.py
```

Puis, pour avoir le descriptif des commandes (aide)

```
> help
```

Ou, pour avoir la liste des commandes (sur 6 colonnes)

```
> ?
```

Pour une installation définitive sur le système référez-vous au fichier INSTALL fourni avec la distribution.

Mentions légales

Auteur

MiKaël NAVARRO <klnavarro@gmail.com> Auteur / concepteur de npí.

Licence

npí est fourni sous licence GPL ², sans aucune garantie de fonctionnement. De même, les corrections de bugs et mises à jour ne sont pas garanties. Référez-vous au fichier LICENCE (qui doit être inclus dans cette distribution) pour plus de détails sur les termes de la licence.

Distribution

<http://freshmeat.net/projects/npí/>

Notes techniques

Algorithme

En notation polonaise inverse, chaque opérateur suit ses opérandes ; une expression infixée comme

```
(1 - 2) * (4 + 5)
```

se note

```
1 2 - 4 5 + *
```

Il n'y a plus besoin de parenthèses ; Cette notation ne présente pas d'ambiguïté, du moment que l'on connaît le nombre d'opérandes que nécessite chaque opérateur.

L'implémentation est simple. On place chaque operande sur une pile ; Lorsqu'un opérateur arrive, on dépile le nombre d'opérandes approprié, on leur applique l'opérateur, et on empile le résultat.

Ainsi dans l'exemple ci-dessus, on empile 1 et 2, puis on les remplace par leur différence, -1. Ensuite, on empile 4 et 5, puis on les remplace par leur somme, 9. Enfin on remplace -1 et 9 par leur produit, -9, sur la pile. Lorsqu'on arrive à la fin de la ligne d'entrée, on extrait la valeur qui se trouve au sommet de la pile et on l'affiche.

La structure du programme est donc une boucle qui réalise l'opération appropriée à chacun des opérateurs et opérandes, au fur et à mesure qu'ils apparaissent

```
tant que (l'operateur ou l'operande suivant
          n'est pas l'indicateur de fin de fichier)
  si (c'est un nombre)
    l'empiler
  sinon, si (c'est un operateur)
    depiler les operandes
    effectuer l'operation
    empiler le resultat
  sinon, si (c'est un caractere de fin de ligne)
    depiler la valeur du sommet et l'afficher
  sinon
    erreur
```

Implémentation

Quelques détails sur l'implémentation :

- La structure de donnée *Stack* (Pile) a été programmé orienté objet (type dérivé de List) ;
- L'utilisation de la libreadline a permis la complétion des noms de commandes ainsi que de simplifier l'implémentation des fonctions.
- Cette calculatrice accepte (depuis la version 0.2.2) des commandes sur la ligne de commande et/ou via un fichier de script.
- Depuis la version 0.3.0, ce logiciel à été migré sur Python 3000.

Outils

IRC bot

Pour m'amuser un peu avec les socket j'ai aussi implémenté un bot IRC primitif

```
$ python ircnpi.py --help
```

Ce dernier se connecte à un serveur IRC (par défaut localhost, càd le démon `ircd-irc2`) sur le canal `#ircnpi` et attend les requêtes des utilisateurs. Ces derniers peuvent ensuite envoyer des commandes au bot (en utilisant un client IRC comme *irssi* ou *erc* sous Emacs) suivant la syntaxe

```
%ircnpi 1 2 - 4 5 + *
```

Et le bot IRCNPI renverra le résultat :)

Pour eteindre le bot

```
%ircnpi stop
```

Jabber bot

Grâce à la librairie [SleekXMPP](#) j'ai aussi créé un bot XMPP qui répond aux calculs qu'on lui propose.

Pour le lancer

```
$ python npibot.py --jid=npibot@klnavarro.fr --password=xxxxxxxx
```

Puis à partir d'un client Jabber (Gajim ou jabber sous Emacs) on ouvre une session de chat avec `npibot`

```
1 2 - 4 5 + *
```

Et le bot renverra le résultat :)

Bibliographie

HP48 Manual: *HP 48G Manuel d'utilisation*. Edition 2.

HP48 Manual: J.-M.Ferrard. *"MATHEZ" la HP 48G/GX*. D3I Diffusion.

-
- 1 <http://www.gnu.org/copyleft/fdl.html>
 - 2 <http://www.gnu.org/copyleft/gpl.html>