

# Nanoshell

`fork()` + `execve()` is all you need.

## Introduction

---

L'objectif de ce projet est d'implémenter un **shell Unix**.



Il y a environ 10.000 features possibles à implémenter dans un shell. Le vôtre n'aura que les features essentielles. Le but est de recoder les fonctionnalités essentielles et de comprendre **comment les fonctions `fork()` / `execve()` / `waitpid()` s'emboîtent !**



# Le projet

---

Voici les fonctionnalités à gérer :

- Quand vous lancez votre programme, vous devez **afficher un prompt**.
- L'utilisateur **saisit une chaîne de caractères** et appuie sur "Entrée" (on ne vous demande pas une gestion avancée de la ligne de saisie : ni déplacement, ni raccourcis, etc... [Curieux ?](#) )
- Si la chaîne décrit un **exécutable présent dans le path**, celui-ci est exécuté avec les paramètres.
- Si la chaîne est un **builtin** reconnu, ce builtin est exécuté avec les paramètres. Builtins à implémenter : `cd`, `exit`, `pwd`, `env`
- Sinon, votre shell doit afficher une erreur :

```
?> grutcblu
nanoshell: weird, grutcblu is not here... :/
```

Concernant `cd`, on vous demande simplement de gérer "`cd ..`", "`cd ~`", "`cd`" tout court, et `cd somedir` (un répertoire du dossier courant, pas de chemin sophistiqué).

## Implémentation

---

**Lecture de la ligne :**

- Vous lisez sur le file descriptor `stdin` (càd 0)
- Vous pouvez découper la ligne par espaces, sans parsing sophistiqué
- Pas de gestion des guillemets requise, ni d'échappement de caractères



- Une ligne vide ne fait rien

### Gestion du PATH :

Si la commande contient un “/”, tenter d’exécuter la commande dans le répertoire courant. Sinon :

- Lire la variable d’environnement PATH
- Tester chaque répertoire qui s’y trouve
- Construire le chemin complet du binaire avec la première “solution” trouvée (ex. : `/usr/local/bin/python3`)

## Aller plus loin

---

- Ctrl-D (EOF) quitte le shell proprement
- Epurage et parsing avant d’exécuter la commande
- Gestion des esperluettes
- [Gestion des pipes](#)
- Gestion des redirections

## Compétences visées

---

- C
- environnement UNIX



## Rendu

---

Le projet est à rendre sur votre github :

<https://github.com/prenom-nom/nanoshell>

## Base de connaissances

---

- <https://man7.org/linux/man-pages/man2/execve.2.html>
- <https://man7.org/linux/man-pages/man2/fork.2.html>
- <https://man7.org/linux/man-pages/man2/wait.2.html>
- <https://man7.org/linux/man-pages/man2/getpid.2.html>
- <https://man7.org/linux/man-pages/man2/chdir.2.html>
- [https://en.wikipedia.org/wiki/Shell\\_builtin](https://en.wikipedia.org/wiki/Shell_builtin)