

EXERCICE ET TRAVAUX PRATIQUES SUR LINUX

COMMANDES UTILISÉES POUR CET EXERCICE

gcc : compilation (GNU projet compilateurs C et C++ (v2.7))

gcc -c : produit uniquement un fichier objet

gcc -o prog -static : produit un fichier exécutable en faisant une liaison statique avec la bibliothèque système

gcc -o prog : idem et prépare une liaison dynamiques(par défaut)

gcc -g : fournit des informations de mise au point en format system natif (stabs, COFF, XCOFF, or DWARF).

(GDB peut fonctionner avec cette information de mise au point)

ldd : imprime les dépendances de la bibliothèque partagée

size : donne les tailles des sections et la taille totale

objdump : fournit les informations disponibles sur les fichiers objets

nm : fournit la liste des symboles issus des fichiers objets

strace : trace les appels systèmes et les signaux

gdb : le débrouilleur GNU ("GNU Debugger")

ps : fournit l'état du processus

sleep : met en attente pour une durée spécifiée

PROGRAMME DE TEST D'ENVIRONNEMENT

```
#include <stdio.h>
```

```
int main(int argn, char *argv[], char *env[])
```

```
{
```

```
    int nb_pas;
```

```
    int i;
```

```
    // for (i=0; env[i] != NULL; i++)
```

```
    for (i=0; i < 6; i++)
```

```
        printf("VARIABLE ENVIRONNEMENT:%s\n", env[i]);
```

```
    if (argn != 2){
```

```
        printf("il faut un argument : le nombre de pas de calcul\n");
```

```
        exit (-2);
```

```
    }
```

```
    sscanf(argv[1], "%d", &nb_pas);
```

```
    printf("nombre de pas = %d \n", nb_pas);
```

```
    sleep(200);
```

```
}
```