

Progetto II - Il C e le chiamate di sistema (1/2)

Scrivere un programma C che prenda come parametro sulla linea di comando il percorso di un file di testo contenente una lista di comandi della shell come nel seguente esempio:

```
ls -l /  
cp file1 file2  
cd; mkdir dir1  
...
```

Il programma C dovrà leggere le linee del file, eseguendo il comando (eventualmente composto) contenuto in ognuna di esse e rispettando le seguenti specifiche:

- il programma dovrà visualizzare l'exit status (valore numerico) del comando appena eseguito,
- il programma, tra l'esecuzione di un comando e quello della linea successiva, dovrà fermarsi attendendo la pressione di un tasto da parte dell'utente prima di proseguire.

Gestire gli errori relativi all'accesso al file di testo e scrivere una breve relazione con alcuni test significativi.

Progetto II - Il C e le chiamate di sistema (2/2)

Suggerimenti:

- per la lettura delle linee del file di testo, utilizzare le funzioni della libreria standard del C (`fopen`, ...),
- per eseguire un comando composto come, ad esempio, `cd; ls -l`, si può utilizzare la seguente chiamata di sistema:

```
execlp("sh", "sh", "-c", buf, NULL);
```

supponendo di aver memorizzato la stringa del comando nel vettore di caratteri `buf`;

- per recuperare l'exit status del processo figlio si può utilizzare una delle seguenti chiamate di sistema:

```
#include <sys/types.h>
```

```
#include <sys/wait.h>
```

```
pid_t wait(int *status);
```

```
pid_t waitpid(pid_t pid, int *status, int options);
```

dove la locazione puntata da `status` conterrà il codice d'uscita del processo figlio.