

Esercizio 1

- Scrivere un programma che prende come parametro un nome di file da creare (`nome_file`)
- Legge i dati da `stdin` e li fa comprimere dall'utility `zip` nell'archivio compresso `nome_file`
- Esempio di comando `bash` che comprime la stringa 'foo bar' immagazzinandola in un un archivio compresso chiamato `nome_file.zip` (l'estensione `.zip` viene aggiunta automaticamente)

```
echo 'foo bar' | zip -q nome_file -
```

Esercizio 2

- Scrivere un programma che prende come parametro un nome di file da creare (nome_file)
- Legge i dati da stdin e li fa comprimere dall'utility zip nell'archivio compresso nome_file
- *Quando l'utente digita Ctrl-C, il programma termina il processo zip, e mostra la dimensione dell'archivio compresso (ovvero lancia il comando `ls -l nome_file.zip`), poi termina.*
- Esempio di comando bash che comprime la stringa 'foo bar' immagazzinandola in un un archivio compresso chiamato nome_file.zip (l'estensione .zip viene aggiunta automaticamente)

```
echo 'foo bar' | zip -q nome_file -
```

Esercizio 3 – due processi

Due processi: A (parent) e B (child), condividono un file aperto in modalità scrittura (e quindi anche l'offset del file).

il processo A fa l'append di una riga di testo al file; dorme un secondo; comunica al processo B di continuare. Il processo A aspetta.

il processo B, a sua volta, fa l'append di una riga di testo allo stesso file; dorme un secondo; comunica al processo A di continuare. Il processo B aspetta.

Quando il processo A ha scritto la centesima riga, manda un segnale di terminazione al processo B, aspetta che B termini e poi termina anche A.

Esercizio 4

ho tre processi A, B e C; simile all'esercizio precedente; i tre processi condividono un file aperto.

A scrive sul file; sleep 1 secondo; avvisa B di procedere

B scrive sul file; sleep 1 secondo; avvisa C di procedere

C scrive sul file; sleep 1 secondo; "tira il dado" per decidere se avvisare il processo A o il processo B di procedere.

Esercizio 5

- Il processo principale origina due processi figli A e B
- Il processo A «produce» un risultato: RA
- Il processo B «produce» un risultato: RB
- Il processo principale aspetta la conclusione dei due processi A e B; quando entrambi hanno concluso l'esecuzione, concatena RA e RB e scrive il risultato sull'output.

Utilità - scaricare un file da pc sul proprio cellulare via internet

Creare una cartella, metterci un file;

dalla cartella di cui sopra, lanciare il comando:

```
python -m SimpleHTTPServer 8000
```

Iscriversi a <https://ngrok.com> (free)

Scaricare l'applicazione ngrok

«connect your account» (ngrok authtoken xyz.....)

Eseguire:

```
ngrok http 8000
```

Accedere all'url riportato da ngrok.

Attenzione alla sicurezza!