

## PROGETTO n.1/2

### COME CONSEGNARE IL PROGETTO:

Consegnare l'esercizio via email a

damiano.perri[at]unipg.it oppure tramite Unistudium

L'oggetto della email deve essere "Progetto 1/2 Cognome Matricola Sistemi Operativi"

Verificare che, una volta inviata la email, questa non venga cancellata o rifiutata da qualche firewall

Spesso le email che contengono zip con codici shell vengono automaticamente cestinate, al caso vi segnali questo

errore inviate di nuovo la email dove lo zip invece di contenere file sh, contiene solo file txt

(nel senso che rinominate l'sh in txt), li capisco lo stesso...

### COME REALIZZARE IL PROGETTO:

L'esercizio va realizzato in ambiente Linux (ad esempio Ubuntu) e deve essere eseguibile su terminale Bash

### COME NON REALIZZARE IL PROGETTO:

Ad oggi siamo circondati da IA generative, dai vari chatgpt, ai Bard e via dicendo.

Sono certamente comode, ma non vi fanno apprendere assolutamente nulla e se usate senza saggezza potrebbero essere

addirittura dannose per il vostro percorso di apprendimento.

Voi siete in un percorso che vi deve formare, preparare al lavoro, allenare alla vita come informatici.

Usare questi strumenti per svolgere l'esercizio impedirà certamente al vostro cervello di creare quei collegamenti

neuronal fondamentali a imparare, comprendere, padroneggiare Bash

Non prendete il progetto come un qualcosa "da fare" ma prendetelo come un primo passo per un percorso di apprendimento

### CONSIGLI:

L'esercizio va svolto autonomamente senza cadere nella

trappola che spinge ad usare IA generative.

Usarle, vi portera' certamente a completare l'esercizio ma non imparate niente.

Sfruttate questi anni di studio, rendeteli utili, produttivi, migliorate voi stessi prima di tutto e cercate con tutte le vostre forze di arrivare sempre al risultato.

Non abbattetevi davanti ad un problema che sembra difficile, ragionate come ci ha insegnato Alan Turing:

Se non siamo in grado di risolvere un problema, scomponiamolo in problemi piu' piccoli, risolviamo quelli e poi ricomponiamo le soluzioni per ottenere la soluzione al problema principale !

Mi raccomando !!

In caso di aspetti non specificati nel testo dell'esercizio o in ogni caso in cui si possano configurare diverse scelte alternative nella progettazione della soluzione, effettuate voi le scelte che vi sembrano piu` ragionevoli (giustificandole opportunamente nei commenti al codice).

Ovviamente non e' consentito utilizzare un comando unix/linux pre-esistente che svolge tutto il compito.

Non si puo' usare "cp -r" tanto per capirci...!

Voglio che risolviatelo il problema senza comandi che fanno "tutto per voi"

#### MODALITA` DI CONSEGNA:

La consegna deve avvenire spedendo un archivio compresso.

L'archivio deve chiamarsi "cognome\_matricola.zip"

es: Se il Mario Rossi ha numero di matricola 123456 allora l'archivio si deve chiamare rossi\_123456.zip

All'interno dell'archivio zip deve essere presente un file txt con:

- Nome, cognome, matricola dello studente
- Il/i file con il codice, debitamente commentato, della soluzione dell'esercizio
- Ogni script bash realizzato deve essere descritto da commenti (bastano poche righe)

#### TESTO DELL'ESERCIZIO:

Si realizzi uno script bash che implementi un comando

"aggiorna" che

accetta due parametri.

I due parametri devono essere pathname (assoluti o relativi) di due directory.

Le due directory non devono essere una contenuta nell'altra, devono essere due directory

allo stesso livello di profondita' del file system.

Inoltre i due pathname non devono essere link simbolici.

Lo script verifica questi requisiti. Se i requisiti non sono soddisfatti stampa un messaggio di errore.

Se tutti i requisiti sono soddisfatti, allora il comando esamina tutti i file

regolari contenuti nella prima directory e verifica se esistono nella seconda.

Se un file non esiste lo copia. Se invece esiste, verifica se quello nella

prima directory sia piu' recente; in tal caso lo copia, altrimenti no.

(Notare che le copie avvengono solo in una direzione: un contenuto della

prima directory viene salvato nella seconda, ma non il contrario)

#### PARTE FACOLTATIVA:

Chi vuole puo' realizzare una versione del comando che accetta una o tutte due queste opzioni

-r se indicata opera ricorsivamente (quindi processa anche le sotto-directory, le sotto-sotto-directory, ecc)

-i se indicata opera come descritto ma prima di copiare un file chiede all'utente se vuole copiarlo o no.

#### COME VIENE VALUTATO IL LAVORO:

Terro' conto della correttezza e della completezza della soluzione rispetto ai

requisiti richiesti (quindi se la soluzione non e' completa o non siete riusciti a svolgere completamente tutto quanto richiesto, la consegna potra' comunque avvenire).

Coloro che termineranno anche la parte facoltativa avranno un bonus all'esame pari a 0.5 punti (per questo esercizio)

Un codice NON indentato correttamente non consente di  
ottenere punti bonus