

Prova scritta 20/02/2008

Matricola	1	2	3	4	tot
	/9	/6	/6	/9	/30

Il candidato risolva gli esercizi proposti nel tempo massimo di 3 ore. Non è consentito avvalersi di libri, appunti o altro sussidio. I risultati degli esercizi vanno messi in una cartella, compressa ed inviata all'indirizzo (antonino.staiano@uniparthenope.it) specificando nell'oggetto la stringa "LABSO" seguito da Nome, Cognome, Matricola e Data dell'Esame.

1. 9 punti (SOLO PER CFU 12)

Realizzare un programma C e Posix sotto Linux che, mediante l'ausilio della libreria pthread, decomponga la somma degli elementi di un vettore di interi di dimensione 1000 in quattro somme locali effettuate concorrentemente da altrettanti thread. Si contempli l'uso di una variabile mutex per regolare l'accesso alla variabile globale che conterrà la somma degli elementi del vettore.

1. 9 punti (SOLO PER CFU 10)

Realizzare un programma in C e Posix sotto Linux che, mediante l'uso di pipe, implementi la seguente pipeline:

ls -al | sort -k 6 | more -d

2. 6 punti (PER CFU 12 e 10)

Scrivere un programma in C che esamini il contenuto della directory corrente, verificando per ogni file regolare se la sua estensione è .c. In questo caso, verificarne i permessi di scrittura per il gruppo e visualizzare sullo standard output il contenuto del secondo file di lunghezza minore.

3. 6 punti (PER CFU 12 e 10)

All'interno del file crypt.txt è contenuta una password costituita dai primi 4 caratteri e dagli ultimi 5 caratteri del testo contenuto nel file. Creare uno script che, accettando come parametro di ingresso il file suddetto, individui la password e stampi a video. Una volta trovata la password (1) inserirla in un file denominato password.txt e contarne i caratteri; (2) confrontare il numero di caratteri trovato con quello contenuto nella variabile count=5 e stampare a video un opportuno messaggio che indichi se il confronto è risultato positivo o meno.

File crypt.txt

Weekend di maltempo sulla penisola.

Neve al nord, pioggia sul resto d'Italia.

4. 9 punti (PER CFU 12 e 10)

Scrivere un programma in C e Posix sotto Linux che crea un processo figlio e dopo 5 secondi gli invia il segnale SIGUSR2. Il figlio attende il segnale SIGUSR2 inserisce il contenuto del file password.txt, dell'esercizio precedente, in un buffer e termina. Non appena il figlio termina, il padre manda a se stesso un segnale SIGUSR1 e quando intercetta il segnale legge il contenuto del buffer e lo scrive sullo standard output, quindi attende per un numero di secondi pari alla lunghezza della stringa contenuta nel buffer e termina.