

Ex. 1	
Ex. 2	
Ex. 3	
Ex. 4	
Ex. 5	
Ex. 6	
Tot.	

# Sistemi Operativi

## Compito d'esame

26 Giugno 2013

Matricola \_\_\_\_\_ Cognome \_\_\_\_\_ Nome \_\_\_\_\_

Docente:       Laface       Quer

**Non si possono consultare testi, appunti o calcolatrici. Riportare i passaggi principali. L'ordine sarà oggetto di valutazione.**

**Durata della prova: 75 minuti.**

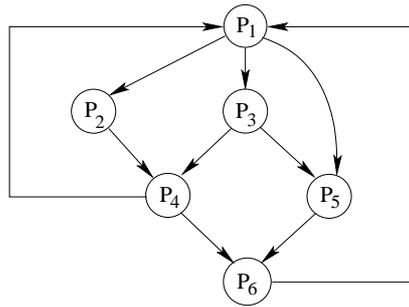
1. Si riporti l'albero di generazione dei processi e si indichi che cosa produce su video il seguente programma e per quale motivo.

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
int main () {
    int i;
    for (i=0; i<2 ; i++) {
        printf ("i=%d\n", i);
        if (fork())          /* call #1 */
            fork();        /* call #2 */
        else
            system ("echo i=i"); /* call #3 */
    }
}
```

2. Si illustrino le caratteristiche delle *pipe* per la comunicazione e la sincronizzazione tra processi. Se ne illustri l'utilizzo con un esempio descritto utilizzando codice in linguaggio C.

3. Si illustri il significato e l'utilizzo dei "grafi di allocazione delle risorse" nel caso di risorse semplici e multiple. Se ne esemplifichi l'utilizzo in presenza e in assenza di deadlock (si richiedono almeno due esempi).

4. Dato il seguente grafo di precedenza, realizzarlo utilizzando il **minimo** numero possibile di semafori. I processi rappresentati devono essere processi ciclici (con corpo del tipo `while(1)`). Si utilizzino le primitive `init`, `signal` e `wait`. Riportare il corpo dei processi ( $P_1, \dots, P_6$ ) e l'inizializzazione dei semafori.



5. Uno script di shell riceve come parametro sulla riga di comando il nome di una directory.

Lo script deve:

- cercare all'interno dell'albero di direttori, specificato dal parametro, tutti i file di testo di estensione `'' .txt ''`, che contengono almeno una riga che incomincia con una cifra numerica (0 ÷ 9).
- cancellare tali righe dai rispettivi file.
- ricopiare i file stessi (senza le righe cancellate) in file con lo stesso nome ma estensione `'' .mod ''`.

Non è consentito utilizzare né AWK né SED.

6. I risultati di un programma nella gestione di diversi progetti sono memorizzati su file. Per ogni riga del file vengono indicati il nome del progetto e i tempi di esecuzione del programma durante tre fasi successive, come di seguito indicato:

```
progA 12.34 26.45 123.99
progB 32.45 16.45 23.23
ex001 56.34 6.45 343.99
pdtsw 112.84 265.45 56.82
```

Si scriva uno script AWK in grado di:

- ricevere il nome di due file del tipo indicato sulla riga di comando.
- visualizzare (a video) l'elenco di tutti i progetti contenuti nei due file, comprensivo del nome del progetto, dei tre tempi di esecuzione, della somma di tali tempi e di una stringa che indica se il progetto compare in entrambi i file oppure in uno solo. I progetti che compaiono in un solo file devono essere individuati dalla stringa "ONE". I progetti che compaiono in entrambi i file devono essere individuati dalla stringa "BEST". In quest'ultimo caso i tempi visualizzati (e la relativa somma) devono essere quelli relativi al (i.e., presenti nel) file per cui la somma dei tempi è minore.

Si osservi che non tutti i progetti sono riportati nei due file e che in ogni caso l'ordine dei progetti nei file non è identico.