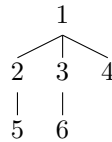


Nume și grupă: _____ Numărul 4

1. (10p) Puteți simula o structură de directoare pe mai multe nivele cu o structură cu un singur nivel? Dacă da, explicați cum ați implementa o astfel de simulare și cum se compară cu structura pe mai multe nivele. Dacă nu, explicați ce vă împiedică să faceți acest lucru. Dați un exemplu.
2. (10p) Fie următoarea arborescență de procese



Scrieți o secvență de cod care reproduce această structură.

3. Fie două procese P_i și P_j care vor să acceseze o zonă critică și următoarea implementare pentru P_i :

```

1  do {
2    flag[i] = true;
3    while (flag[j]) {
4      if (turn == j) {
5        flag[i] = false;
6        while (turn == j)
7          ; /* do nothing */
8        flag[i] = true;
9      }
10   }
11   /* critical section */
12   turn = j;
13   flag[i] = false;
14   /* remainder section */
15 } while (true);

```

unde procesele împart **flag[2]** (inițializat cu **false**) și **turn**.

- (a) (5p) Ce se întâmplă dacă eliminăm linia 13?
- (b) (10p) Arătați dacă soluția satisface cele trei proprietăți: exclusivitate mutuală, progres și timp finit de așteptare.
4. Fie două matrice $A \in \mathbb{N}^{6 \times 6}$ și $B \in \mathbb{N}^{4 \times 6}$ ținute contiguu în memorie pe linii și fie un sistem în care avem 5 *frame*-uri disponibile. În acest sistem într-o pagină încap 6 întregi, iar programele P1 și P2 de mai de jos încap fiecare separat într-o pagină.

```

1  P1:
2  for (i = 0; i < 6; i++)
3    for (j = 0; j < 6; j++)
4      A[i][j] = i+2*j;

1  P2:
2  for (i = 0; i < 4; i++)
3    for (j = 0; j < 6; j++)
4      B[i][j] = i+j;

```

Presupunem că programele se execută concurrent astfel: fiecare program stă pe procesor cât să ducă până la capăt instrucțiunea de la linia 4 o singură dată după care cedează locul concurrentului.

- (a) (5p) Cum arată programele și datele repartizate pe pagini?
- (b) (5p) Cum arată diagrama Gantt folosind algoritmul LRU cu o strategie de înlocuire a paginilor globală (în care ambele programele pot folosi toate *frame*-urile disponibile în sistem).
- (c) (5p) Cum ați aloca *frame*-urile pentru o strategie de înlocuire a paginilor locală (în care fiecare program primește un număr fix de *frame*-uri pe care doar el le poate utiliza). De ce?