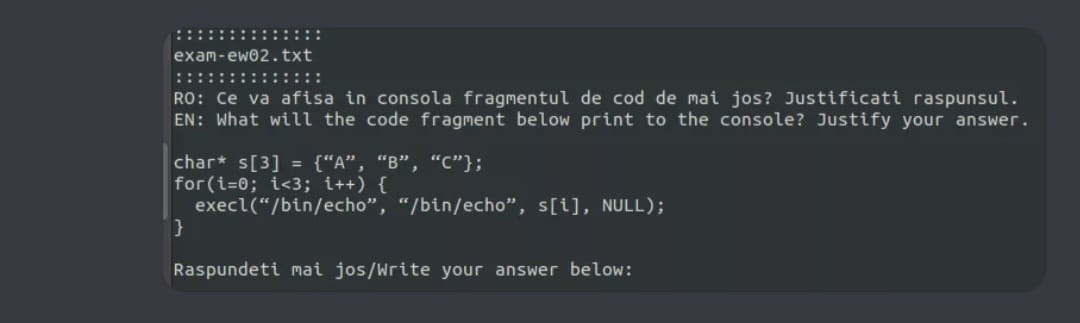
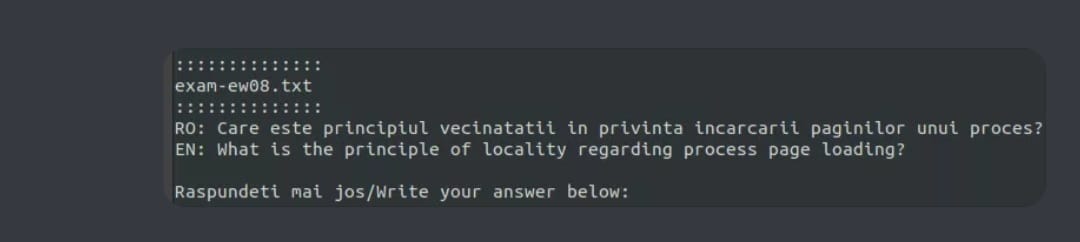


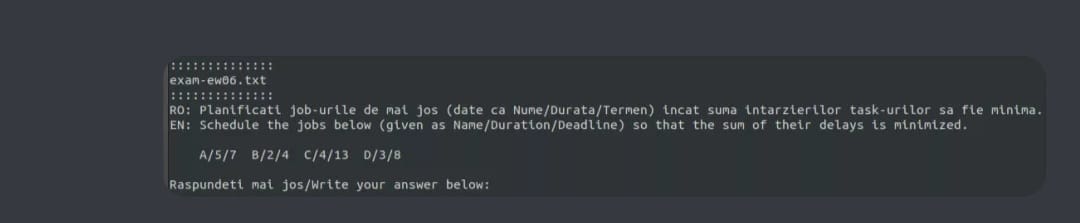
Grep -E ‘([aeiou].\*){3,}’ fisier



Afiseaza A



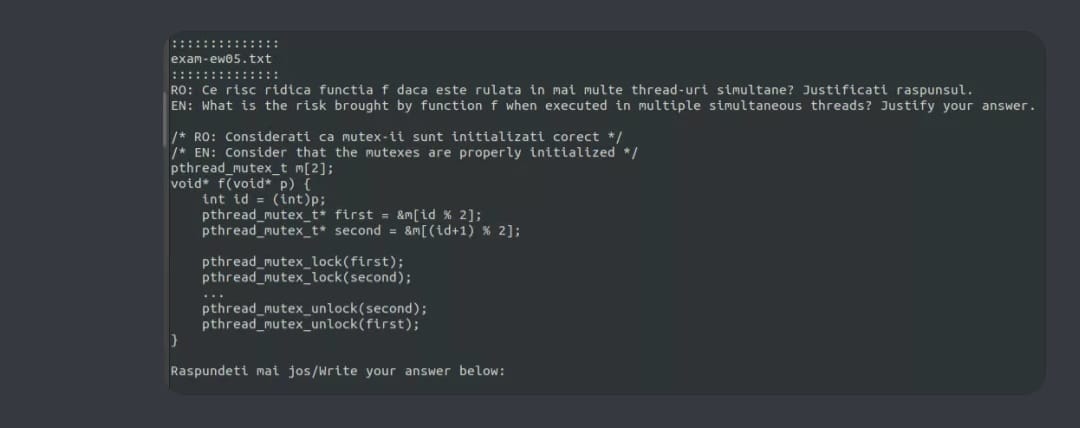
Cand incarcam o pagina refetita, incarcam si cateva pagini de langa ea.



B A D C

2 + 5 +3 + 4

O intarziere de 2 la D si una la C . intarziere totala de 3

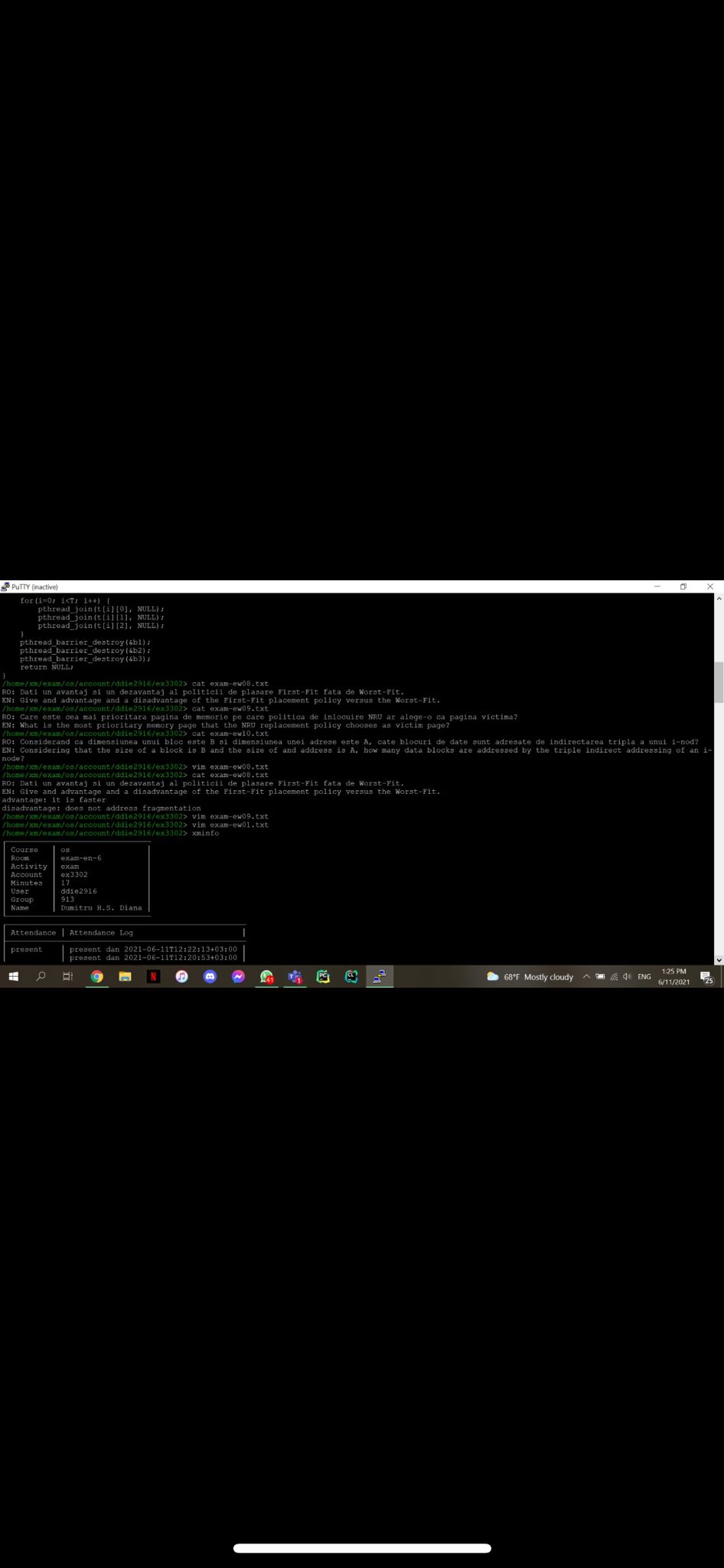


Deadlock

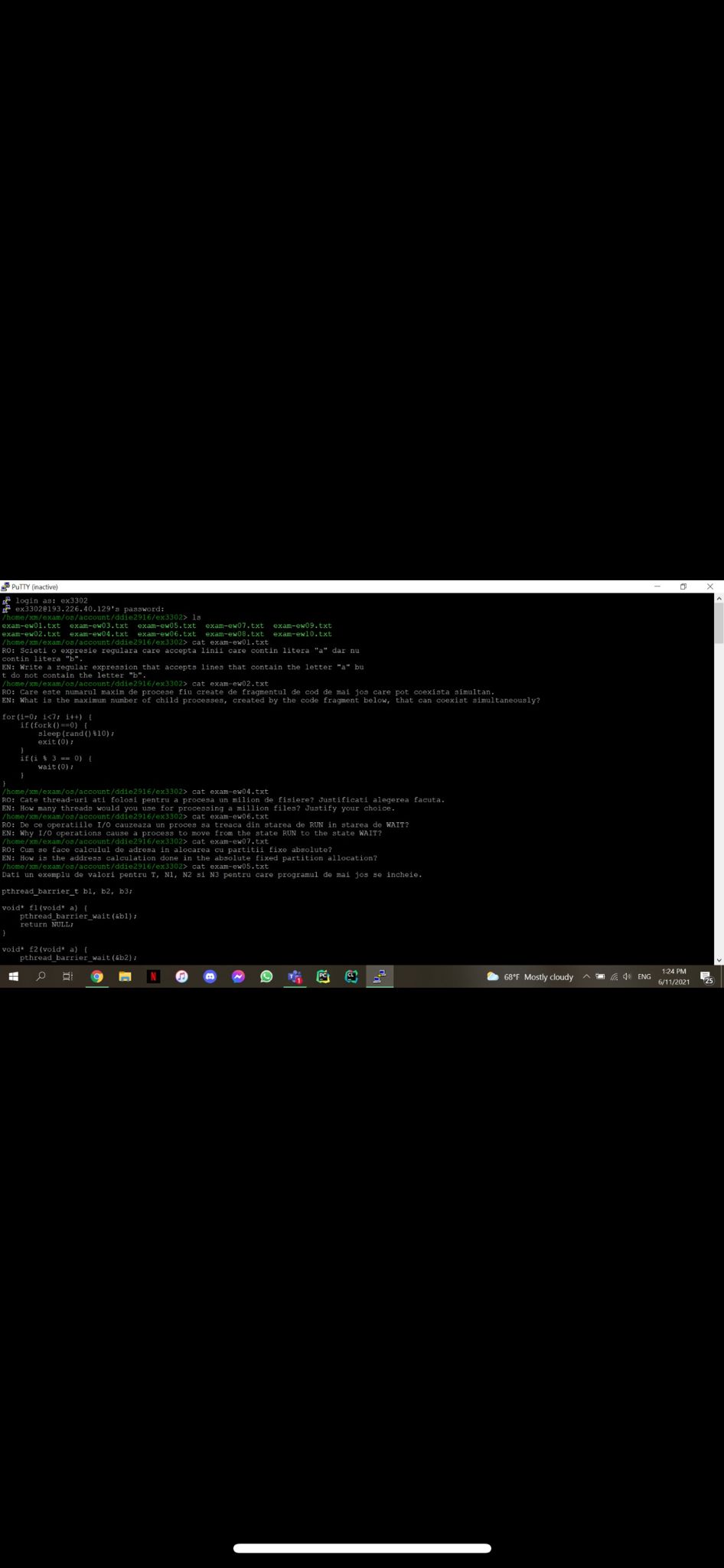
Lock(0) lock(1)

Lock(1) lock(0)

Si apuca fiecare thread sa faca primul lock.



1. First Fit e mai rapida dar nu adreseaza fragmentarea
2. Se allege prima clasa nevida
3. (B/A)^3



Grep -E -I ‘[a]{1,}’ fisier | grep -E -I -v ‘[b]{1,}’ fisier

Grep -E -I ‘^[^b]\*a[^b]\*$’ fisier

2. 7-3 = 5 procese ce pot coexista

3. Daca de exemplu am 4 core uri as porni de la 4 threaduri si asa stiu ca lucreaza in parallel pe core uri diferite. Masor timpul de executie. Si ma duc cu multiplii de 4 pana cand timpul de rulare creste si se allege nr de threaduri pentru care am obtinut cel mai mic timp de executie.

4. Pentru ca dureaza mult si ca sa nu tina ocupat procesorul le punem in wait pana se incarca

5. Adresa executabilului e aceeasi cu adresa fizica.

6. Valori egale



14

In A

Open de write pt X

Open de read pt Z

In b

Open de write pt Y, si read pt X

In C open de read pt Y, write pt Z si

