

Test practic la SO – varianta nr. 3 – Barem de corectare

Observații generale:

- cele trei programe C, scriptul și fișierul de intrare trebuie plasate într-o ierarhie de directoare conform celor descrise în enunțul problemei (și apelate în mod corespunzător poziției lor în ierarhia respectivă); de asemenea, conținutul arhivei finale trebuie să păstreze ierarhia respectivă de directoare și fișiere sursă.
- fiecare dintre cele trei programe C va fi compilat în același (sub)director în care se află programul sursă, denumind executabilul cu același nume ca și sursa, dar fără extensia .c
- pentru implementarea comunicațiilor între cele trei procese cooperante se vor utiliza exact metodele de comunicare descrise în specificații.
- fiecare punctaj din barem se acordă **integral, sau deloc**.

1. Baremul pentru scriptul "starter.sh":

- a) 0.25p – scriptul va verifica primirea parametrului p , conform punctului i) din specificația dată.
- b) 0.5p – scriptul va porni mai întâi execuția în *background* a programului (specificat cu calea către subdirectorul corespunzător) indicat în enunț: **analyzer**, dacă $p=0$, respectiv **service**, dacă $p=1$.
- c) 0.25p – scriptul va face o pauză de 1 secundă sau, respectiv, 2 secunde, în funcție de valoarea lui p .
- d) 0.5p – scriptul va porni și execuția celui alt program (specificat cu calea către subdirectorul corespunzător) indicat în enunț: **service**, dacă $p=0$, respectiv **analyzer**, dacă $p=1$.
Notă: b)+d) – în ambele situații, apelul programului **analyzer** trebuie să conțină un argument în linia de comandă: calea către fișierul de intrare "input.txt", conform punctului iv) din specificația dată.
- e) 0.25p – scriptul va aștepta terminarea execuțiilor celor trei programe.
- f) 0.25p – scriptul va afișa conținutul mapării nepersistente cu nume.
- g) 2 x 0.25p – scriptul va șterge maparea nepersistentă cu nume și, respectiv, fișierul fifo.

Total : 2.5p pentru scriptul "starter.sh"

2. Baremul pentru programul "analyzer.c":

- a) 0.25p + 0.25p – verificarea primirii unui singur argument la linia de comandă + testarea dreptului de citire, conform punctului i) din specificația dată.
- b) 1p – implementarea corectă a inițializării și configurării acelei mapări nepersistente cu nume, pentru a putea comunica, în ambele sensuri, cu procesul **service**, conform punctului i) din specificația dată.
- c) 0.25p – implementarea corectă a citirii din fișierul de intrare, în funcția descrisă la punctul ii) din specificația dată.
- d) 0.5p – implementarea corectă a extragerii cuvintelor din conținutul fișierului de intrare, în funcția descrisă la punctul ii) din specificația dată.
- e) 0.5p + 0.5p – implementarea corectă și, respectiv, fără *bug-uri* de tipul *race-condition* sau *segmentation fault*, a transmiterii rezultatelor către procesul **service** prin acea *mapare nepersistentă cu nume*, în funcția descrisă la punctul ii) din specificația dată.
- f) 0.5p + 0.5p – implementarea corectă și, respectiv, fără *bug-uri* de tipul *race-condition* sau *segmentation fault*, a recepționării datelor de la procesul **service** prin acea *mapare nepersistentă cu nume*, în funcția descrisă la punctul iii) din specificația dată.
- g) 0.25p – afișarea datelor recepționate de la procesul **service**.

Notă: se va calcula fracția procentuală de îndeplinire a criteriilor de mai sus, iar cele trei criterii de mai jos vor fi notate prin ponderare cu acest procentaj: $F = \text{total puncte acordate pentru criteriile a)-g)} / 4.5p$

e) $F * 1p$ - tratarea cazurilor de eroare la TOATE apelurile de sistem din program.

f) $F * 1p$ - programul se compilează fără erori și fără warning-uri.

g) $F * 0.5$ - scrierea codului într-o formă indentată, plăcută ochiului, i.e. "readable code" (de exemplu, evitați copy-paste în mcredit).

Total: 7p pentru programul "analyzer.c"

3. Baremul pentru programul "service.c":

- a) 0.25p + 0.5p – implementarea corectă a operațiilor fork și, respectiv, exec descrise la punctul i) din specificația dată.
- b) 0.25p + 0.5p – implementarea corectă a creării unui canal anonim și, respectiv, a configurării comunicației prin canalul anonim pentru a putea primi date de la procesul **worker**, conform punctului ii) din specificația dată.
- c) 0.25p + 0.25p – implementarea corectă a creării unui canal fifo și, respectiv, a configurării comunicației prin canalul fifo pentru a putea trimite rezultate către procesul **worker**, conform punctului ii) din specificație.
- d) 1p – implementarea corectă a inițializării și configurării acelei mapări nepersistente cu nume, pentru a putea comunica, în ambele sensuri, cu procesul **analyzer**, conform punctului iii) din specificația dată.
- e) 0.5p + 0.5p – implementarea corectă și, respectiv, fără *bug*-uri de tipul *race-condition* sau *segmentation fault*, a recepționării rezultatelor de la procesul **analyzer** prin acea *mapare nepersistentă cu nume*, în funcția descrisă la punctul iv) din specificația dată.
- f) 0.5p – implementarea corectă a procesării informațiilor citite, în funcția descrisă la punctul iv) din specificația dată.
- g) 0.75p – implementarea corectă (și fără *bug*-uri de concurență) a transmiterii datelor către procesul **worker** prin acel *canal anonim*, în funcția descrisă la punctul iv) din specificația dată.
- h) 0.75p – implementarea corectă (și fără *bug*-uri de concurență) a recepționării rezultatelor de la procesul **worker** prin acel *canal fifo*, în funcția descrisă la punctul v) din specificația dată.
- i) 0.5p – implementarea corectă a procesării datelor primite de la procesul **worker**, în funcția descrisă la punctul v) din specificația dată.
- j) 0.5p + 0.5p – implementarea corectă și, respectiv, fără *bug*-uri de tipul *race-condition* sau *segmentation fault*, a transmiterii informațiilor procesate către procesul **analyzer** prin acea *mapare nepersistentă cu nume*, în funcția descrisă la punctul v) din specificația dată.

Notă: se va calcula fracția procentuală de îndeplinire a criteriilor de mai sus, iar cele trei criterii de mai jos vor fi notate prin ponderare cu acest procentaj: $F = \text{total puncte acordate pentru criteriile a)-j)} / 7.5p$

x) $F * 1p$ - tratarea cazurilor de eroare la TOATE apelurile de sistem din program.

y) $F * 1p$ - programul se compilează fără erori și fără warning-uri.

z) $F * 0.5$ - scrierea codului într-o formă indentată, plăcută ochiului, i.e. "readable code" (de exemplu, evitați copy-paste în mcredit).

Total: 10p pentru programul "service.c"

4. Baremul pentru programul "worker.c":

- a) 0.5p – implementarea corectă a configurării comunicației prin canalul anonim pentru a primi date de la procesul **service**, conform punctului i) din specificația dată.
- b) 0.25p – implementarea corectă a configurării comunicației prin canalul fifo pentru a trimite date către procesul **service**, conform punctului i) din specificația dată.
- c) 0.75p – implementarea corectă (și fără *bug*-uri de concurență) a recepționării informațiilor de la procesul **service** prin acel *canal anonim*, în funcția descrisă la punctul ii) din specificația dată.
- d) 0.75p – implementarea corectă a selecției trigramelor care vor fi transmise procesului **service**, în funcția descrisă la punctul ii) din specificația dată.
- e) 0.75p – implementarea corectă (și fără *bug*-uri de concurență) a transmiterii rezultatelor către procesul **service** prin acel *canal fifo*, în funcția descrisă la punctul ii) din specificația dată.

Notă: se va calcula fracția procentuală de îndeplinire a criteriilor de mai sus, iar cele trei criterii de mai jos vor fi notate prin ponderare cu acest procentaj: $F = \text{total puncte acordate pentru criteriile a)-e)} / 3p$

x) $F * 1p$ – tratarea cazurilor de eroare la TOATE apelurile de sistem din program.

y) $F * 1p$ – programul se compilează fără erori și fără warning-uri.

z) $F * 0.5p$ – scrierea codului într-o formă indentată, plăcută ochiului, i.e. "readable code" (spre exemplu, evitați copy-paste în mcredit).

Total: 5.5p pentru programul "worker.c"

Total general 1. - 4. : 25p