

**Teză la disciplina**  
**Arhitectura calculatoarelor și sisteme de operare**  
**Nr. 1**

1. Pe care dintre magistralele unui sistem de calcul sunt transmise codurile instrucțiunilor pe care le execută procesorul?
2. Se consideră un sistem cu 32 Mo de memorie, cu dimensiunea paginii de 4 Ko (1 Ko = 1024 octeți). Două procese aflate simultan în memorie necesită 10620 octeți, respectiv 8230 octeți. Dintre acestea, 1000 octeți reprezintă o zonă de memorie partajată de cele două procese (se regăsește în tabelele de paginare ale ambelor procese). Câtă memorie se pierde prin fragmentare internă?
3. Dacă un procesor are un pipeline de 22 stagii, iar verificarea condiției de salt se face în stagiul 16, câte cicluri de ceas se pierd în cazul unei predicții greșite?
4. Se consideră un alocator în care la un moment dat există următoarele zone libere:
  - patru zone de 64 Ko
  - o zonă de 128 Ko
  - două zone de 512 Ko
  - două zone de 1 Mo (aflate în aceeași pagină)
  - trei zone de 2 MoApare o cerere pentru o zonă de dimensiune 256 Ko. Care va fi configurația zonelor libere după servirea acesteia, dacă alocatorul este de tip:
  - a) alocator cu puteri ale lui 2
  - b) alocator McKusick-Karels, cu dimensiunea paginii de 2 Mo
5. Prin ce diferă arhitectura RISC de cea CISC? Ce avantaje prezintă prima față de cea de-a doua?
6. Fie un procesor dotat cu memorie cache, având următoarele caracteristici:
  - timpul de acces la cache:  $T_c = 2.5 \text{ ns}$
  - timpul de acces la memorie în cazul unei ratări în cache:  $T_m = 20 \text{ ns}$
  - timpul de acces la memoria principală în absența cache-ului:  $T_p = 18 \text{ ns}$Care trebuie să fie rata de succes a cache-ului pentru ca timpul mediu de acces la memorie în prezența cache-ului să fie 75% din timpul de acces la memorie fără cache?
7. Care sunt criteriile după care este apreciată performanța unui alocator de memorie?
8. Procesoarele RISC au un număr mare de regiștri de uz general, pentru a reduce numărul de accese la memorie. În condițiile sporului de performanță adus de utilizarea cache-ului, mai este justificată această abordare?
9. Descrieți părțile componente ale unui fișier obiect.
10. Un proces aflat în execuție este scos din această stare la terminarea perioadei de timp alocate (chiar dacă nu a generat nici o eroare și nu a solicitat efectuarea nici unui apel sistem), pentru a ceda locul altui proces. În general, toate procesele au alocate perioade de execuție egale. Ar putea fi justificată ideea ca unele procese să aibă alocate perioade de execuție mai lungi decât altele?

**Teză la disciplina**  
**Arhitectura calculatoarelor și sisteme de operare**  
**Nr. 2**

1. Descrieți părțile componente ale unui fișier obiect.
2. Se consideră un alocator în care la un moment dat există următoarele zone libere:
  - patru zone de 64 Ko
  - o zonă de 128 Ko
  - o zonă de 1 Mo
  - trei zone de 2 MoApare o cerere pentru o zonă de dimensiune 512 Ko. Care va fi configurația zonelor libere după servirea acesteia, dacă alocatorul este de tip:
  - a) alocator cu puteri ale lui 2
  - b) alocator McKusick-Karels, cu dimensiunea paginii de 4 Mo
3. Care sunt criteriile după care este apreciată performanța unui alocator de memorie?
4. Se consideră un sistem cu 64 Mo de memorie, cu dimensiunea paginii de 2 Ko (1 Ko = 1024 octeți). Două procese aflate simultan în memorie necesită 10620 octeți, respectiv 8230 octeți. Dintre acestea, 2500 octeți reprezintă o zonă de memorie partajată de cele două procese (se regăsește în tabelele de paginare ale ambelor procese). Câtă memorie se pierde prin fragmentare internă?
5. De ce registrul PC (contorul program) nu poate fi modificat direct prin program?
6. Dacă un procesor are un pipeline de 16 stagii, iar verificarea condiției de salt se face în stagiul 13, câte cicluri de ceas se pierd în cazul unei predicții greșite?
7. Prin ce diferă arhitectura RISC de cea CISC? Ce dezavantaje prezintă prima față de cea de-a doua?
8. Fie un procesor dotat cu memorie cache, având următoarele caracteristici:
  - timpul de acces la cache:  $T_c = 3 \text{ ns}$
  - timpul de acces la memorie în cazul unei ratări în cache:  $T_m = 25 \text{ ns}$
  - timpul de acces la memoria principală în absența cache-ului:  $T_p = 23 \text{ ns}$Care trebuie să fie rata de succes a cache-ului pentru ca timpul mediu de acces la memorie în prezența cache-ului să fie 60% din timpul de acces la memorie fără cache?
9. Un proces aflat în execuție este scos din această stare la terminarea perioadei de timp alocate (chiar dacă nu a generat nici o eroare și nu a solicitat efectuarea nici unui apel sistem), pentru a ceda locul altui proces. În general, toate procesele au alocate perioade de execuție egale. Ar putea fi justificată ideea ca unele procese să aibă alocate perioade de execuție mai lungi decât altele?
10. Capacitatea totală a memoriei cache este produsul dintre numărul de linii de cache și dimensiunea unei linii. Pentru o memorie cache de capacitate totală dată, este de preferat să avem linii de cache cât mai multe sau cât mai mari?

**Teză la disciplina**  
**Arhitectura calculatoarelor și sisteme de operare**  
**Nr. 3**

1. Se consideră un sistem cu o memorie de 175000 octeți, care folosește segmentarea memoriei. La un moment dat, tabelul descriptorilor de segment arată astfel (s-au reprezentat doar segmentele alocate):

Indice	1	2	5	7	8
Adresă start	25000	1000	150000	111000	83000
Dimensiune	24000	14000	15000	23000	25000

Mai trebuie alocate două segmente, de 9000 octeți, respectiv 30000 octeți (în această ordine). Dintre algoritmi de plasare a segmentelor în memorie, care reușesc plasarea ambelor segmente și care eșuează? Nu se face compactarea memoriei.

2. Dacă un procesor are un pipeline de 19 stagii, iar verificarea condiției de salt se face în stagiul 15, câte cicluri de ceas se pierd în cazul unei predicții greșite?

3. Ce limitări prezintă modul utilizator al procesorului și de ce au fost introduse aceste limitări?

4. În ce situații este de preferat legarea dinamică a unei funcții în locul legării statice?

5. Ce acțiuni realizează editorul de legături cu ajutorul informațiilor din fișierele obiect?

6. Ce este o dependență structurală într-un sistem cu pipeline? Dați un exemplu.

7. Se consideră un alocator McKusick-Karels, într-un sistem cu dimensiunea paginilor de 64 Ko (1 Ko = 1024 octeți). La un moment dat, configurația zonelor libere este următoarea:

- trei zone de 32 octeți
- o zonă de 256 octeți
- șase zone de 512 octeți
- zece zone de 1 Ko
- șapte zone de 4 Ko
- cinci zone de 8 Ko

În ce condiții poate fi servită o cerere pentru o zonă de memorie de 32 Ko?

8. Un procesor are o memorie cache de 256 Ko, cu următoarele caracteristici:

- rata de succes:  $H = 90\%$
- timpul de acces la cache:  $T_c = 4 \text{ ns}$
- timpul de acces la memorie în cazul unei ratări în cache:  $T_m = 20 \text{ ns}$
- timpul de acces la memoria principală în absența cache-ului:  $T_p = 18 \text{ ns}$

Mai există două variante ale procesorului, cu cache de 512 Ko și respectiv 1 Mo. La fiecare dublare a capacității cache-ului, rata de insucces se înjumătățește, în schimb timpul de acces la cache și timpul de acces la memorie în cazul unei ratări în cache cresc fiecare cu câte 1.5 ns. Care variantă de cache este mai rapidă?

9. Un proces aflat în execuție este scos din această stare la terminarea perioadei de timp alocate. Este posibil ca unele procese să aibă alocate perioade de execuție mai lungi decât altele. În acest caz, ceasul de timp real ar trebui să genereze întreruperi la intervale neregulate de timp?

10. Dacă primul Ko de memorie ar fi ocupat de memorie ROM, adresele rutinelor de tratare ale întreruperilor nu ar putea fi modificate. Din punct de vedere al sistemului de operare, această abordare ar fi avantajoasă sau dezavantajoasă?

**Teză la disciplina**  
**Arhitectura calculatoarelor și sisteme de operare**  
**Nr. 4**

1. Un procesor are o memorie cache de 256 Ko, cu următoarele caracteristici:

- rata de succes:  $H = 95\%$
- timpul de acces la cache:  $T_c = 2 \text{ ns}$
- timpul de acces la memorie în cazul unei ratări în cache:  $T_m = 20 \text{ ns}$
- timpul de acces la memoria principală în absența cache-ului:  $T_p = 19 \text{ ns}$

Mai există două variante ale procesorului, cu cache de 512 Ko și respectiv 1 Mo. La fiecare dublare a capacității cache-ului, rata de insucces se înjumătățește, în schimb timpul de acces la cache și timpul de acces la memorie în cazul unei ratări în cache cresc fiecare cu câte 2 ns. Care variantă de cache este mai rapidă?

2. Ce acțiuni realizează editorul de legături cu ajutorul informațiilor din fișierele obiect?

3. Ce este o dependență de date într-un sistem cu pipeline? Dați un exemplu.

4. Se consideră un sistem cu o memorie de 200000 octeți, care folosește segmentarea memoriei. La un moment dat, tabelul descriptorilor de segment arată astfel (s-au reprezentat doar segmentele alocate):

Indice	1	2	5	7	8
Adresă start	25000	1000	150000	111000	83000
Dimensiune	24000	14000	10000	23000	25000

Mai trebuie alocate două segmente, de 20000 octeți, respectiv 35000 octeți (în această ordine). Dintre algoritmi de plasare a segmentelor în memorie, care reușesc plasarea ambelor segmente și care eșuează? Nu se face compactarea memoriei.

5. Se consideră un alocator McKusick-Karels, într-un sistem cu dimensiunea paginilor de 16 Ko (1 Ko = 1024 octeți). La un moment dat, configurația zonelor libere este următoarea:

- patru zone de 32 octeți
- o zonă de 256 octeți
- două zone de 512 octeți
- zece zone de 1 Ko
- trei zone de 4 Ko

În ce condiții poate fi servită o cerere pentru o zonă de memorie de 8 Ko?

6. La apariția unei întreruperi de orice tip, procesorul trece din modul utilizator în modul nucleu. La terminarea rutinei de tratare se revine în modul utilizator. De ce se procedează astfel?

7. Dacă un procesor are un pipeline de 24 stagii, iar verificarea condiției de salt se face în stagiul 18, câte cicluri de ceas se pierd în cazul unei predicții greșite?

8. În ce situații este de preferat legarea dinamică a unei funcții în locul legării statice?

9. Dacă ar exista un singur tabel de paginare global, în loc de a avea câte un tabel pentru fiecare proces, s-ar ocupa mai puțin spațiu cu aceste structuri. Ce dezavantaj ar avea o asemenea soluție?

10. Un proces aflat în execuție este scos din această stare la terminarea perioadei de timp alocate. Este posibil ca unele procese să aibă alocate perioade de execuție mai lungi decât altele. În acest caz, ceasul de timp real ar trebui să genereze întreruperi la intervale neregulate de timp?