

Според која формула се одредува искористеноста на CPU?

Кај мултипрограмен систем кој опслужува n процеси и секој од нив чека на В/И за дел од своето време, искористеноста на ЦПУ се одредува според следната формула: $\eta_{CPU} = 1 - p_n$

Кои својства треба да ги има критичната секција?

1. Взаемна исклучивост - најмногу еден процес по критичен сегмент
2. Прогрес - најмалку еден процес при постоечко барање на критичен сегмент
3. Ограничено чекање – ниту еден процес не чека „премногу“

Што е взаемна исклучителност?

Ако еден процес работи во критичен сегмент, ниеден друг не смее да работи во тој критичен сегмент, т.е. во даден критичен сегмент можеме да имаме најмногу еден процес во еден момент.

Што е прогрес?

Ако нема процес што моментално работи во критичен сегмент, а има процеси што сакаат да работат во критичен сегмент, тогаш еден од нив добива дозвола.

Што е ограничено чекање?

Мора да има граница на бројот на обиди кога други процеси имаат дозвола да влезат во критичниот сегмент по кои процесот што има барање да влезе во критичен сегмент ќе биде услужен.

Што е алгоритам на стареење?

Алгоритмот на стареење го пресметува следното предвидување како тежинска сума од претходните две извршувања, т.е. ако T_0 и T_1 се две последователни времиња на извршување, тогаш T_2 се пресметува како:

$$T_2 = aT_0 + (1-a)T_1$$

Што е натпревар на процеси?

Натпревар меѓу два или повеќе процеси е ситуација кога процесите читаат или запишуваат во заеднички мемориски простор, а крајниот резултат зависи од тоа точно кој од процесите прв ќе добие пристап до заедничкиот ресурс.

Која е разликата помеѓу зафатено чекање (busy waiting) и блокирање (blocking)?

- Кај зафатено чекање процесот постојано проверува вредност на некоја променлива се додека таа не добие одредена вредност.

- Кај блокирање, процесот се блокира и не може да се извршува се додека одреден настан не се случи. За да продолжи со работа, потребно е друг процес да го “разбуди”.

Што е ќорсокак?

Ќорсокак е ситуација кога повеќе процеси кои чекаат на ресурси за да продолжат, никогаш да не ги добијат тие ресурси бидејќи нив ги “држат” исто така процеси кои чекаат.

Кој е потребен и доволен услов за ќорсокак?

- Може да се докаже дека ако графот нема циклуси, тогаш нема ќорсокак.
- Циклус е потребен услов за ќорсокак.
- Ако секој ресурс има само по една инстанца, тогаш е и доволен.

Кои се услови за ќорсокак?

За да дојде до ќорсокак потребно е да се исполнети истовремено следните 4 услови:

1. взаемно исклучување - само еден процес може истовремено да користи еден ресурс.
2. задржи и чекај - мора да постои процес кој држи ресурс и чека на друг зафатен ресурс .
3. нема прекинување - употребата на ресурсот не може да се прекине, само процесот што го држи може да го ослободи.
4. кружно чекање - мора да постои множество $\{P_0, P_1, \dots, P_n\}$ на процеси кои чекаат на ресурси меѓу себе.
 P_0 чека за ресурс кој го држи P_1 , P_1 чека за ресурс кој го држи P_2 и P_n чека за ресурс кој го држи P_0 .

Кои се методи за справување со ќорсокак?

- Превенција
- Избегнување
- Откривање и опоравување
- Игнорирање

Што е превенција?

- Со превенција се обезбедува дека никогаш нема да дојде до ќорсокак.
- Тоа може да се реализира така што ќе се обезбеди барем еден од 4-те услови да не се исполни.

Што е избегнување?

- Секој процес треба да ги декларира своите максимални побарувања. Состојбата на алокација на ресурси се дефинира преку бројот на расположиви и алоцирани ресурси и бројот на побарувањата.
- Состојба е сигурна од ќорсокак ако може да се изврши доделување на ресурсите на сите процеси во некаков редослед.

Што е откривање и справување?

- Ако ОС нема механизми за заштита од ќорсокак тогаш може да дојде до него.
- Потребно е системот да обезбеди механизми за откривање на ќорсокак и опоравување на системот.

Што е игнорирање?

- Механизмите за превенција и избегнување на ќорсокак се скапи и бараат голема комплексност на ОС.
- Кај некои системи се применува игнорирање на ќорсокак, бидејќи е проценето дека таква појава се случува многу ретко и не се исплаќа да се вградуваат заштитни механизми (пр. UNIX).

Банкаров Алгоритам?

1. Нека **Work := Available** и **Finish[i] := false**, $i=1, \dots, n$.

2. Најди i за кое

Finish[i] = false

Need[i] <= Work

Ако не постои такво i , оди на 4.

3. **Work := Work + Allocation_i**

Finish[i] := true

Оди на 2.

4. Ако **Finish[i] := true** за секое i , тогаш системот е сигурен од ќорсокак.

Resource request алгоритам?

- Нека **Request_i[j] = k** ако процесот P_i побарува k единици од ресурсот P_j

- **Resource request** алгоритмот е следниот:

1. Ако **Request_i <= Need_i**, оди на 2 инаку пријави грешка
2. Ако **Request_i <= Available**, оди на 3. Во спротивно P_i мора да чека
3. Нека системот се однесува како да ги доделил бараните ресурси и ги извршил следните измени:
 - **Available := Available - Request**
 - **Allocation := Allocation + Request**
 - **Need_i := Need_i - Request_i**
4. Ако состојбата што се добива е сигурна, додели ги ресурсите. Ако не, врати ја старата состојба.

Откривање на ќорсокак?

- За ресурси со само една единица се дефинира алгоритам за откривање на ќорсокак кој користи Wait – for граф
- Wait – for граф се добива од **resource allocation graph** така што се отстрануваат јазлите со ресурси и ребрата.
- Потоа се поставува ребро $P_i \rightarrow P_j$ ако важи $P_i \rightarrow R_q \wedge R_q \rightarrow P_j$
- Ако овој граф има циклус, значи системот е во ќорсокак.
- ОС треба да одржува ваков граф и периодично да проверува дали има циклуси.
- Не може да се примени кога има повеќе инстанци од ист тип на ресурси.

Опоровување од ќорсокак?

1. Да се информира операторот.

2. Системот сам да се справи со ќорсокакот.

- Проблеми:

- прекин на процес(и)
 - Прекини ги сите процеси во ќорсокак
 - Прекинувај по еден процес, се додека не се прекине ќорсокакот.
- одземање на ресурс
 - Кој да биде жртва?
 - Дали може да се направи rollback?
 - Дали може да се обезбеди дека нема да дојде до изгладнување?

Адресата генерирана од CPU – логичка адреса(корисничките програми работат со логичка адреса)

- Адреса видена од мемориската единица (она што се вчитува во меморискиот адресен регистар) - физичка адреса
- Логичката адреса за време на извршувањето на програмата е виртуелна адреса
- Множество од сите логички адреси генерирани од програмата – логички (виртуелен) адресен простор
- Множество од сите физички адреси што одговараат на соодветните логички адреси – физички адресен простор
- Хардверскиот уред **Memory Management Unit (MMU)** прави мапирање од виртуелна во физичка адреса (во рун-тима)

Структура на датотеки

- Поле (Field) – основен податочен елемент
- Запис (Record) – колекција од поврзани записи кои може да се третираат како единка
- База (Database) – колекција од сврзани податоци и операции дефинирани врз нив
- Датотека (File) – колекција од слични записи. Се третира како единка од корисниците и се контактира преку одделено име. Контрола на пристап е најчесто на датотечко ниво

Алокациски методи за управување со секундарна меморија

- Непрекината алокација
- Поврзана алокација
- Индекцирана алокација

Хардверот на еден В/И уред се состои од 4 делови:

- Магистрала, преку која уредот комуницира со CPU-то (се дели меѓу повеќе уреди)
- Порта(Port), типично се состои од 4 регистри:
 - статус (зафатен? податоци спремни? грешка?)
 - контрола (команда за извржување)
 - податок во (податок што се праќа од уредот кон CPU-то)
 - податок од (податок што се праќа од ЦПУ-то кон уредот)
- Контролер, што прима команда од магистралата, ја трансформира во акција на уредот и чита податоци од и запишува податоци на магистралата
- Самиот уред

Ротационо задоцнување (латенција): време за кое почетокот на секторот доаѓа до главата

Време на пристап: време на барање + ротационо задоцнување

Време на трансфер на податоците:

$$T = b / (r * N)$$

каде T е време на трансфер

b е број на бајти кои треба да се пренесат

r е ротациона брзина во револуции во секунда

N е број на бајти на траката

Кој се улуслови за блокада?

1. Услов за взаемно исклучување. Секој ресурс е или доделен на точно еден процес или е слободен.
2. Услов за hold и wait. Процесите што моментно држат ресурси, доделени претходно можат да побаруваат нови ресурси.
3. Услов за одземање на ресурси. Ресурси претходно доделени на процес, не можат да бидат присилно одземени од него. Тие можат да бидат ослободени само од процесот кој ги држи
4. Услов за кружно чекање. Мора да постои кружен ланец од 2 или повеќе процеси, кои чекаат на ресурс зафатен од следниот процес во синџирот.

Што е внатрешна и надворешна фрагментација?

- Внатрешна кога имаме поделена меморија на блокови со иста или различна големина и кога ќе се пополни меморијата но нема да го пополни цел блок, ќе остане некој дел празен
- Надворешна на пример кога ќе влезат повеќе проеси и еден од тие ќе се изврши и ќе заврши ја ослободува и тука тоа останув празно место.

Што е непрекинатата алокација? За што се користи?

- Непрекинатата алокација кога цело време се сместуваат еден до друг блоковите од меморија, и се користи за сместување на блокови на меморија на диск.
- Наједноставна алокациска шема во која датотеките се сместуваат во континуирани сектори. Овој приод има 2 големи предности. Првата е едноставниот начин на работа при што за локација на секторот потребни се само 2 броја: диск адресата на првиот сектор и бројот на сектори во датотеката. Така бројот на било кој сектор може да се најде просто со собирање.

Генерализирај ги карактеристиките на главните политики за распределување на процеси?

- Сите тие имаат некои параметри според кои се распоредува извршувањето на процесите, на пример Round Robin според временски квантум, FCFS според тоа кој прв пристигнал, SJN според работата која треба да ја извршат, имаат некое време на чекање, средно време на чекање

Што се монитори? За што служат?

- множество на процедури, променливи и податочни структури групирани во специјален вид модули, многу слични на објектите во Јава, Ц++ и содржат општи променливи, условни променливи, функции, и конструктор. Процесите може да го повикаат мониторот во секој момент но не можат да пристапат до неговите податочни структури со процедури декларирани надвор од мониторот.. Се користат како решение за имплементација на конкурентност и синхронизација.
- условните променливи имаат две операции:
 1. x.wait - процесот кој ја повикува оваа операција се блокира (чека) додека некој друг процес не ја сигнализира променливата со x.signal
 2. x.signal - Деблокира некого што чека на променливата

Што се семафори? За што служат?

- Семафор е нов тип променлива која може да има вредност 0 што индицира дека не се случиле wake-up-ови и некоја позитивна целобројна вредност ако се случиле повеќе wake-up-ови. Се користат како решение за имплементација на конкурентност и синхронизација.

- Постојат 2 операции со семафори: **down** и **up** - генерализација на заспивањата и будењата соодветно.

- **Down** операцијата првин проверува дали вредноста на семафорот е поголема од 0. Ако е ја декрементира и продолжува. Ако е 0, тогаш процесот се заспива без да ја заврши down операцијата во тој момент. Проверувањето на вредноста, менувањето и можното заспивање се одвива во една атомично т.е. додека се одвива ниту еден друг процес нема достап до семафорот.

- **Up** операцијата ја инкрементира вредноста на семафорот. Ако 1 или повеќе процеси спијат, или биле заспани од тој семафор, неможејќи да ја довршат претходно нивната down-операција, еден од нив рандом избран од системот добива дозвола да си ја доврши својата операција.

- Семафорите се состојат од:

- променлива **v**
- листа на процеси **L**

Што е политика со испразнување, а што без испразнување?

- со политика на испразнување ќе се прекине тековниот процес ако новодојдениот има повисок приоритет.

- со политика без испразнување ќе се постави новодојдениот процес на чело на редицата на спремни процеси

Дали ако настане изгладнување се подразбира дека има блокада, образложи?

- Може да дојде до изгладнување на процеси но тоа не значи дека ќе дојде до блокада. Изгладнување е кога некој процес чека многу долго да добие право на извршување, најчесто поради понизок приоритет, но не мора да значи дека ќе се случи блокада. На пример кај SJN може некој процес кој има подолга работа и да чека на извршување, но не значи дека ќе настане блокада, другите процеси се извршуваат.

Параметри кои го одредуваат времето на пристап до делови од диск?

- Ротационо задоцнување (латенција), време на пристап, време на трансфер на податоците

- вкупно просечно време на пристап е: $T_a = T_s + 1 / (2 * r) + b / (r * N)$

Што е изгладнување?

- Изгладнување е кога на еден процес во мултипрограмска околина постојано му се одзема правото за извршување (добивање на ресурси), можхе да дојде до изгладнување заради постојано блокирање поради приоритет и може многу да чека на ЦПУ, и можно е системот да падне и процесот никогаш да не се изврши.

Наведи неколку предности, можности од користење на цели мемориски блокови.

Нишките во различни процеси можат да се извршуваат конкурентно.

Дефинирај меморија со директен пристап(ДМА). Напиши примери каде таа би била полезна.

- Меморија со директен пристап е систем кој може да ја контролира меморискиот систем без да го користи ЦПУ за време на пренос туку го користи само на почеток и крај на трансферот.

Примери: при големи трансфери на податоци (диск, ЦД, ДВД) каде е скапо да биде ангажиран процесорот.

Која е предноста од користење на Round Robin распределувачки алгоритам? Кои се недостатоците при негово користење?

- предности од користење на Роунд Робин е тоа што е едноставен и лесен за имплементација и нема опасност од појава на стареење на процесите и процесите се извршуваат по редослед без некаков приоритет, во одреден временски квантум обично дефиниран од 10 до 100 милисекунди (time - sharing)

- Недостатоци при користење на Роунд Робин е тоа што со користеше на овој алгоритам се губи многу повеќе време од колку со другите и некој побитен процес да се заврши подоцна од некој друг што е со помал приоритет.

Како функционира индексирана датотека? За што примарно се користи?

- Индексната датотека функционира така што запишува секвенца според клуч (не само еден), има два типа на индекси, има еден главен клуч кој го имаат сите записи, а записите не мора да бидат со иста големина, и се користи во апликации каде времето мора да е мало и каде исцрпувачко пребарување не се бара.

Што е натпревар на процеси?

- Тоа е ситуација меѓу два или повеќе процеси кога процесите читаат или запишуваат во заеднички мемориски простор, а крајниот резултат зависи од тоа кој од процесите прв ќе добие пристап до заеднички ресурс.

Што е ќорсокак?

- Во мултипрограмска околина може да дојде до ситуација повеќе процеси да чекаат на ресурси за да продолжат но никогаш да не ги добијат бидејќи се зафатени од други процеси кои исто така чекаат, оваа ситуација се вика ќорсокак.