SO Curs 4

Planificarea executiei

Implementarea planificării

Kernel preemptiv
Planificarea in Windows
Planificarea in Linux

Planificarea in sisteme batch

Criterii sisteme batch
First Come First Served
Shortest Job First
Shortest Remaining Time First

Noțiuni de planificare

Starea proceselor
Comportamentul proceselor
Schimbarea de context
Planificarea executiei
Apelarea planificatorului
Implementarea planificatorului

Planificarea in sisteme interactive

Sisteme interactive
Round Robin
Planificarea cu prioritati
Shortest Process Next
Planificarea pentru sisteme real-time

Criterii de planificare

Cooperativ si preemptiv
Timpi de planificare
Criterii de planificare
Tipuri de planificare
Notatii

Cuprins

Notiuni de planificare
Criterii de planificare
Planificarea executiei in sisteme batch
Planificarea pentru sisteme interactive
Planificarea pentru sisteme real-time
Implementarea planificarii

Suport de curs

OSCE
- Capitolul 5 – CPU Scheduling

MOS
- Capitolul 2 – Processes und Threads
- Instalarea 7 - Scheduling

Cuvinte cheie

stările procesului
context switch
CPU-bound
IO-bound
planificarea executiei
planificatori/scheduler
algoritmi de planificare
criterii de planificare
echitate (fairness)
predictibilitate (throughput)
timp de așteptare
turnaround time
procese batch
procese interactive
procese real-time
FCFS
Round Robin
cuanta de timp
prioritate
CFS

Prezi

SO Curs 4

Planificarea executiei

Implementarea planificării

Kernel preemptiv
Planificarea in Windows
Planificarea in Linux

Planificarea in sisteme batch

Criterii sisteme batch
First Come First Served
Shortest Job First
Shortest Remaining Time First

Noțiuni de planificare

Starea proceselor
Comportamentul proceselor
Schimbarea de context
Planificarea executiei
Apelarea planificatorului
Implementarea planificatorului

Planificarea in sisteme interactive

Sisteme interactive
Round Robin
Planificarea cu prioritati
Shortest Process Next
Planificarea pentru sisteme real-time

Criterii de planificare

Cooperativ si preemptiv
Timpi de planificare
Criterii de planificare
Tipuri de planificare
Notatii

Cuprins

Notiuni de planificare
Criterii de planificare
Planificarea executiei in sisteme batch
Planificarea pentru sisteme interactive
Planificarea pentru sisteme real-time
Implementarea planificarii

Suport de curs

OSCE
- Capitolul 5 – CPU Scheduling

MOS
- Capitolul 2 – Processes und Threads
- Instalarea 7 - Scheduling

Cuvinte cheie

stările procesului
context switch
CPU-bound
IO-bound
planificarea executiei
planificatori/scheduler
algoritmi de planificare
criterii de planificare
echitate (fairness)
predictibilitate (throughput)
timp de așteptare
turnaround time
procese batch
procese interactive
procese real-time
FCFS
Round Robin
cuanta de timp
prioritate
CFS

Prezi

Suport de curs

OSCE

- Capitolul 5 – CPU Scheduling

MOS

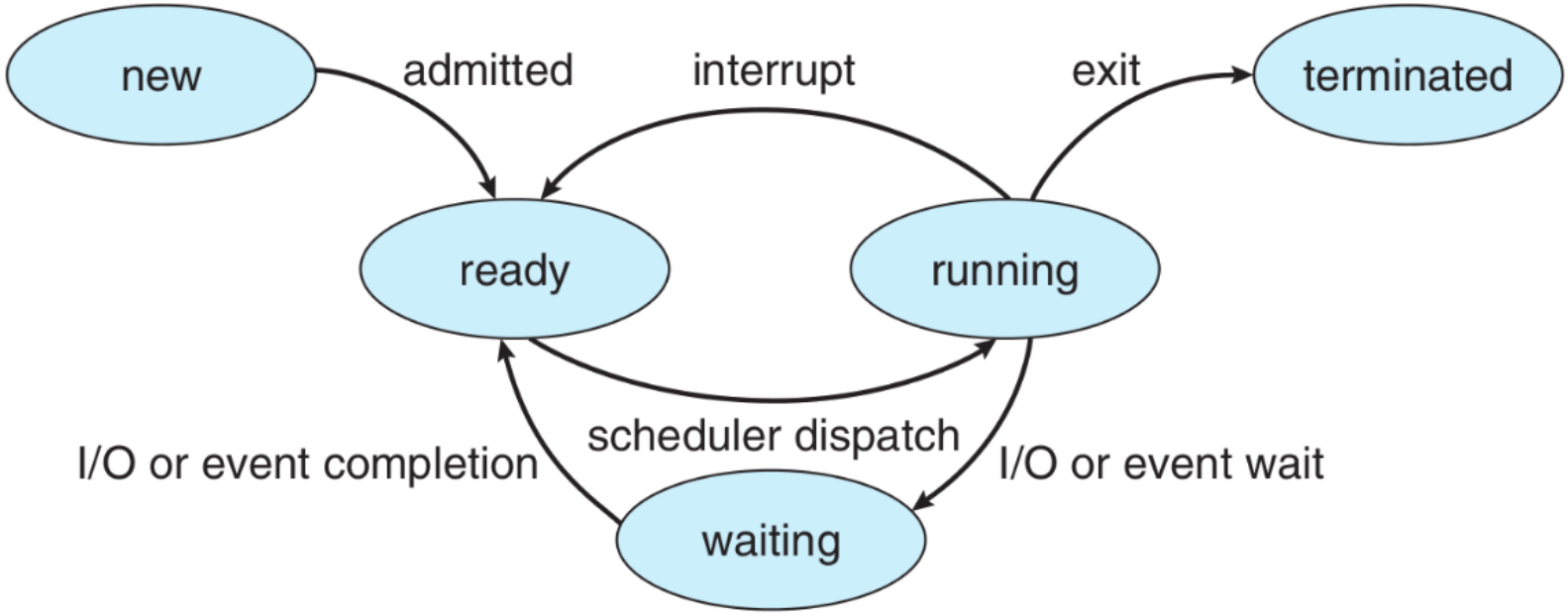
- Capitolul 2 – Processes and Threads
  - Sectiunea 5 - Scheduling

Cuprins

Noțiuni de planificare

Starea proceselor
Comportamentul proceselor
Schimbarea de context
Planificarea executiei
Apelarea planificatorului
Implementarea planificatorului

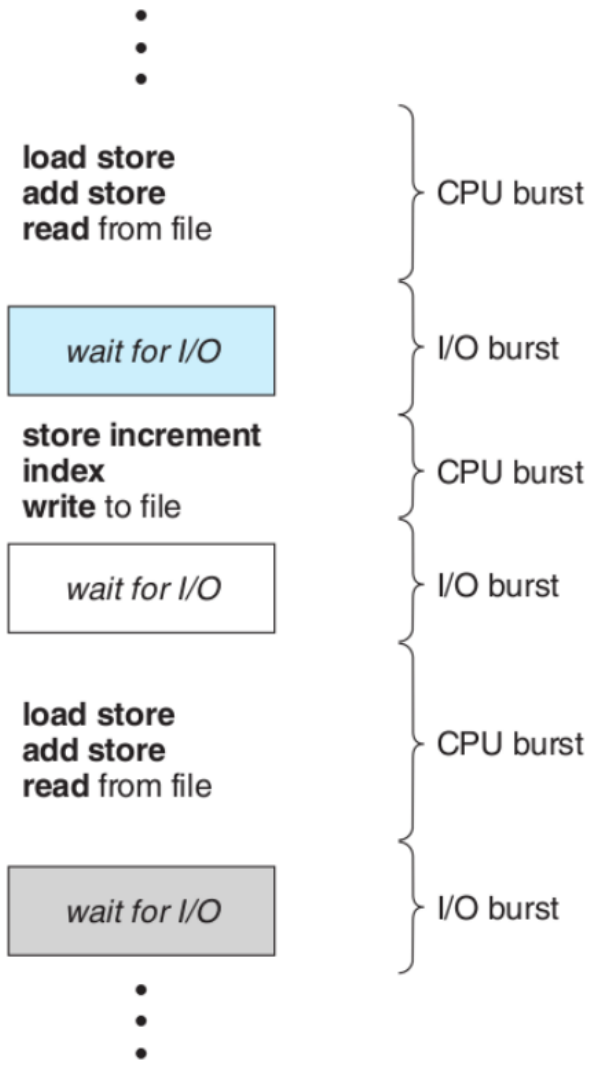
Starea proceselor



in marea parte a timpului procesele sunt in starea waiting
in starea running sunt cel mult N procese

- N = numarul de core-uri

Comportamentul proceselor



Comportamentul proceselor (2)

CPU-intensive
IO-intensive

CPU bursts
I/O bursts

tranzitii intre stari
schimbare de context

Schimbarea de context

context switch

trecerea unui proces din READY in RUNNING
cel din RUNNING trece in READY sau WAITING

overhead al schimbarii de context

- salvarea contextului curent
- incarcarea noului context
- asteptarea incheierii unui apel de sistem

Planificarea executiei

inlocuirea unui proces cu un alt proces
- un context cu un alt context

eficienta: un proces blocat nu tine procesorul ocupat
echitate: un proces este intrerupt pentru a da voie altuia

se apeleaza planificatorul
- alegerea unui proces
- inlocuirea procesului curent

Apelarea planificatorului

când procesul din starea RUNNING moare

cand procesul din starea RUNNING se blocheaza
- operatie blocanta (apel de sistem)

cand procesului din starea RUNNING ii expira cuanta
- intreruperea de ceas

cand un proces READY e prioritar

Implementarea planificatorului

o functie apelata in urma
- unui apel de sistem (se blocheaza procesul, moare procesul)
- unei intrerupere (expira cuanta)

functia se cheama schedule() sau dispatch()

alte functii actualizeaza atributele proceselor si sorteaza procesele in functie de criterii de planificare

planificatorul (functia schedule()):
- salveaza contextul curent (registre, resurse)
- selecteaza un proces
- incarca un context nou

Criterii de planificare

Cooperativ si preemptiv
Timpi de planificare
Criterii de planificare
Tipuri de planificare
Notatii

nizare

Cooperativ si preemptiv

voluntary/involuntary preemption

cooperativ
- yielding
- da acces voluntar procesorului
- interactivitate scazuta
- implementare simpla

preemptiv
- procesul este preemptat
- de obicei expira cuanta
  - intrerupere de ceas
- interactivitate sporita
- de avut in vedere sincronizare

Timpi de planificare

timp de asteptare: timp de asteptare in READY
turnaround time: timp de rulare pe ceas

- de la intrarea in sistem pana la iesirea din sistem

dorim timpi cat mai mici

- timp de asteptare mic: sistem interactiv
- turnaround time mic: sistem productiv

in general nu poti avea si sistem productiv si interactiv

Criterii de planificare

gradul de ocupare a procesorului
- cat mai mare

productivitate (throughput)
- numar de procese incheiate
- cat mai mare

fairness
- toate procesele sa aiba acces la procesor/resurse

(mean) turnaround time
- cat mai mic

timp (mediu) de raspuns
- intervalul de la intrarea in sistem pana la rulare prima oara
- cat mai mic

timp (mediu) de asteptare
- cat mai mic

Tipuri de planificare

planificarea sistemelor batch (background processing)
- accentul pe productivitate

planificarea sistemelor interactive
- accentul pe interactivitate/fairness

planificarea proceselor real-time
- indeplinirea sarcinii in timp util

Notatii

WT - Waiting Time
MWT - Mean Waiting Time
TT - Turnaround Time
MTT - Mean Turnaround Time
J - job (batch processing)
P - process (interactive processing)

Planificarea in sisteme batch

Criterii sisteme batch
First Come First Served
Shortest Job First
Shortest Remaining Time First

Criterii sisteme batch

throughput
turnaround time
utilizarea procesorului

First Come First Served
Shortest Job First
Shortest Remaining Time Next

First Come First Served

FCFS

planificare in ordinea intrarii in sistem
un proces care cere procesorul este trecut într-o coada de asteptare
procesele care se blocheaza sunt trecute la sfarsitul cozii

\+ usor de inteles si implementat
\- procesele CPU-bound incetinesc procesele I/O-bound
  • convoying
\- timp mediu de asteptare/turnaround destul de mare

J1, J2, J3
joburile intra simultan in sistem
timpii de executie: 24, 3, 3

FCFS: ordinea J1, J2, J3
TT(J1) = 24; TT(J2) = 27; TT(J3) = 30
MTT = (24 + 27 + 30) / 3 = 27

Shortest Job First

SJF

se planifica jobul cel mai scurt
- trebuie cunoscuta durata de executie

J1, J2, J3, J4
job-urile intră simultan în sistem
timpii de execuie: 12, 20, 8, 4

FCFS: J1, J2, J3, J4
TT(J1) = 12; TT(J2) = 32; TT(J3) = 40; TT(J4) = 44
MTT = (12 + 32 + 40 + 44) / 4 = 32

SJF: J4, J3, J1, J2
TT(J4) = 4; TT(J3) = 12; TT(J1) = 24; TT(J2) = 44
MTT = (4 + 12 + 24 + 44) / 4 = 21

Shortest Remaining Time First

SRTF

trebuie cunoscut timpul de executie a jobului

versiune preemptiva a algoritmului SJF

- cand un nou job este submis pentru executie
- ...si timpul de executie al acestuia este mai mic decat timpul ramas din executia jobului curent
- jobul curent este suspendat si noul job este executat

SRTF: J1(0:1), J2(1:5), J4(5:10), J1(10:17), J3(17:26)
TT(J1) = 17; TT(J2) = 4; TT(J3) = 24; TT(J4) = 7
TTM = (17 + 4 + 24 + 7) / 4 = 13

J1, J2, J3, J4
timpii de intrare tn sistem: 0, 1, 2, 3
timpii de executie: 8, 4, 9, 5

SJF: J1(0:8), J2(8:12),J4(12:17),J3(17:26)
TT(J1) = 8; TT(J2) = 11; TT(J3) = 24; TT(J4) = 14
TTM = (8 + 11 + 24 + 14) / 4 = 14.25

Planificarea in sisteme interactive

Sisteme interactive
Round Robin
Planificarea cu prioritati
Shortest Process Next
Planificarea pentru sisteme real-time

Sisteme interactive

sisteme desktop
e importanta interactiunea cu utilizatorul

timpul de raspuns
interactivitate
fairness

Round Robin
clase de prioritati
Shortest Process Next

Round Robin

time sharing

FCFS preemptiv
cuanta de timp de rulare a programului
la expirarea cuantei de timp procesul este preemptat

cuanta de timp mare
- productivitate ridicata
- interactivitate reduse

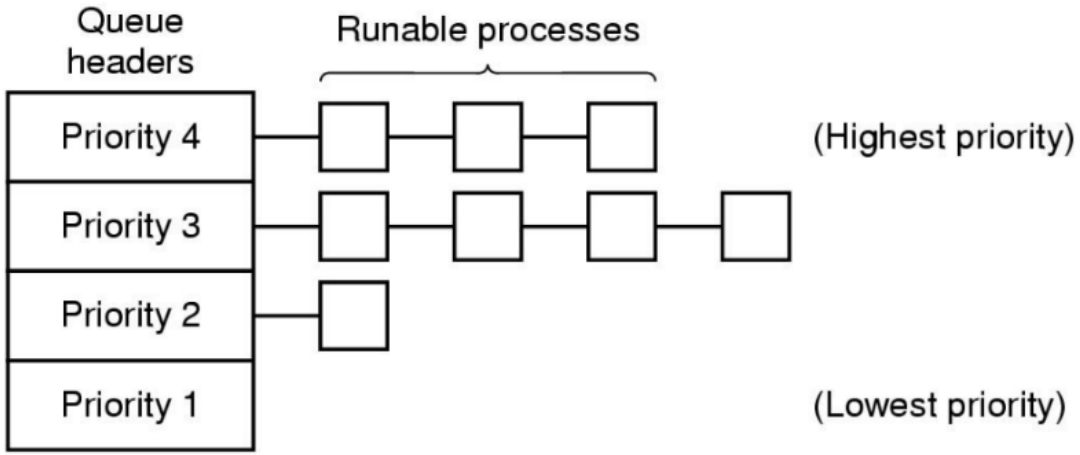
cuanta de timp mica
- interactivitate sporita
- productivitate redusa
  - timp consumat in schimbari de context

Planificarea cu prioritati

dezavantaj Round-Robin
- toate procesele sunt „egale”

abordarea pentru planificare cu prioritati
- unele procese sunt „mai egale” decat altele
- utilizatori importanti/mai putin importanti
- există procese mai importante/prioritare

prioritati dinamice si statice



Shortest Process Next

adaptare a SJF pentru sisteme interactive

problema: nu se cunoaste timpul de executie

solutie

- estimare pe baza comportamentului anterior
- se estimează o durată T0
- procesul durează T1
- estimarea pentru următoarea cuantă va fi a * T1 + (1-a) * T0
- a – estimarea se uită sau nu repede
- tehnică de estimare de tip aging

Planificarea pentru sisteme real time

criterii importante
- indeplinirea operatiilor in timp limitat
- predictibilitatea

hard real-time
- rezervarea resurselor
- nu se foloseste swapping sau memorie virtuală

soft real-time
- procesele critice au prioritate maximă
- pot cauza întârzieri mari celorlalte procese

Linux/Windows au implementare de soft real-time

Implementarea planificării

Kernel preemptiv
Planificarea in Windows
Planificarea in Linux

Kernel preemptiv

un proces poate fi intrerupt in timp ce lucreaza in kernel space

spatiul kernel e comun tuturor proceselor
- un proces poate fi intrerupt in orice punct din kernel
- nevoie de locking

latenta redusa, sistem mai responsiv

dificultate sporita in implementare (locking)

Linux si Windows au kernel preemptiv

Planificatorul in Windows

algoritm de planificare preemptiva bazat pe prioritati
subsistemul de planificare se cheamă dispatcher

o schemă de prioritati pe 32 de niveluri

- 0 – prioritate sistem
- 1-15 – clasă variabila de prioritati
- 16-30 – clasă real-time

nucleu preemptiv

Algoritmul de planificare in Windows

este selectat procesul cu prioritatea cea mai mare
fiecare prioritate are asociata o coada

procese din fiecare coada sunt procese in starea READY

procesele din aceeasi coada sunt planificate asemanator cu Round-Robin

Planificarea executiei in Linux

planificator preemptiv, time-sharing
suport pentru procese real-time

kernel preemptiv de la versiunea 2.6

planificatorul gestionează 4 clase de procese [5]

- real-time FCFS
- real-time RR
- interactive
- batch

Completely Fair Scheduler

CFS

2.6.23 – prezent

time-ordered red-black tree

virtual runtime
- fiecare proces are un timp „virtual” de executie
- cel mai din stanga proces are timpul virtual cel mai mic
  - va fi urmatorul planificat

planificare: t_virtual_curent – t_virtual_stanga > threshold

operatii in O(log n)

Planificarea proceselor real-time

conform cu POSIX.1b

FCFS

- un proces din această clasa va fi inlocuit doar daca efectuează o operatie blocanta

RR

- la fel ca FCFS, dar un proces va fi preemptat daca ii consuma cuanta de timp
- cuanta de timp este mai mare decat pentru procesele interactive

Cuvinte cheie

starea proceselor
context switch
CPU-bound
IO-bound
planificarea executiei
planificator/scheduler
algoritmi de planificare
criterii de planificare
echitate (fairness)
productivitate (throughput)
timp de asteptare
turnaround time
procese batch
procese interactive
procese real-time
FCFS
Round Robin
cuanta de timp
prioritate
CFS