UPRAVLJANJE S PROCESI

(Seminarska naloga)

Mentor:

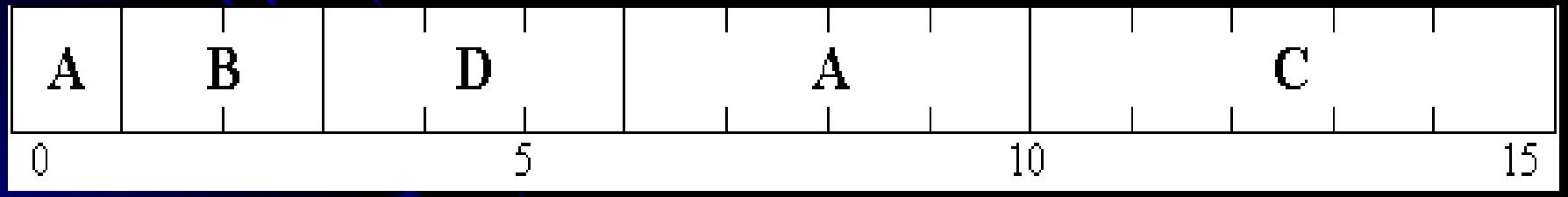
Avtor:

Beleženje uspešnosti algoritma

- Uspešnost algoritma se da analizirati, če so nam podani primerni podatki o procesu
- Za Najkrajši Preostali Čas potrebujemo podatke o tem, kdaj se začne proces izvajati in koliko časa posamezen proces potrebuje da se izvede.
- Čakalni čas se lahko izračuna tako, da od celotnega časa izvajanja odštejemo čas, ki je potreben za izvedbo procesa.

Tabela – podatki o izvajanju

Proces	Začetek izvajanja	Trajanje procesa
A	0.0	5
B	1.0	2
C	2.0	5
D	3.0	3



Lastnosti procesa

- OS shrani informacije o procesu v ***Procesov Nadzorni Blok*** (PCB):
  - stanje procesa in podatki o razvrščanju,
  - podatki o upravljanju s pomnilnikom,
  - informacije o delovanju strojne opreme,
  - informacije o signalizaciji procesa,
  - informacije o kontroli dostopa do podatkov
  - informacije o dodeljenih sredstvih
  - V/I sistem
  - accounting information.

STANJE PROCESA IN RAZVRŠČANJE

- OS mora biti sposoben v vsakem trenutku hitro prepoznati vse procese v določenem stanju.
- ločeni seznami ali vrste za vsako stanje
- Informacije, ki jih uporabljajo algoritmi za razvrščanje se nahajajo v PCB-ju.

UPRAVLJANJE S POMNILNIKOM

- proces mora imeti dostop samo do tistih pomnilniških naslovov, kateri mu pripadajo.
- Za vsak proces mora OS ohranjati kopijo nastavitev upravljanja s pomnilnikom.
- Ko proces dobi nadzor nad CPU, se register postavi v stanje, ki je zapisano v nastavitvah za upravljanje s pomnilnikom, ali pa se nastavitvene vrednosti prekopirajo v poseben pomnilnik upravljalnika z registri.

STANJE STROJNE OPREME

- Na večprogramskih sistemih, ko proces dobi nadzor nad CPU-jem, se mora več strojnih registrov postaviti na vrednosti, ki so jih imeli, ko je bil proces odstavljen.
- OS shraniti vrednosti teh spremenljivk za vsak proces, ki ni v fazi izvajanja.
- Odvisno od sestave strojne arhitekture, lahko podatek vsebuje različne vrednosti

STANJE STROJNE OPREME

- podatek vsebuje vrednosti:
  - programskega števca,
  - akumulatorskih registrov,
  - glavnih registrov,
  - kazalcev sklada,
  - indeksnih registrov,
  - stanje aritmetično logične enote in
  - stopnjo prioritete izvajanja.
- Resetiranje vseh vrednosti, ko nov proces dobi nadzor nad procesorjem, se imenuje ***context-switch.***

SIGNALIZACIJA PROCESOV

- OS lahko podpira neko obliko signalizacije procesov.
- Signal je navidezna prekinitev, ki jo sproži OS.
- Iz operacijskega sistema ali iz procesa.
- Enemu ali več procesov.
- Proces, ki sprejme signal, se lahko nanj odzove tako, da ga:
  - ignorira,
  - ga konča,
  - izvede zahtevano operacijo.

SIGNALIZACIJA PROCESOV

- Lahko imamo tudi predpis, ki določi da bo signal poslan ob določenem času v prihodnosti.
- **Programski časovniki** so signali, ki jih proces pošlje sam sebi ob določeni točki v prihodnosti.
- OS mora navadno tudi vzdrževati seznam vseh signalov, ki so že bili poslani, vendar še niso bili obdelani.

KONTROLA DOSTOPA

- Dva tipa informacij o kontroli dostopa sta lahko shranjena:
  - informacija, ki ugotovi dostopne pravice procesa.
  - informacije o primarnih dostopnih pravicah, ki so pripisane kateremu koli objektu, ki ga ta proces ustvari.

- Odvisno od uporabljenega kontrolnega mehanizma, so te informacije gesla, zmožnosti ali seznami kontrole dostopa.

VHODNO/IZHODNI SISTEM

- OS mora vzdrževati informacije o datotekah in napravah, ki jih uporablja nek proces ali so na voljo procesu preko Vhoda/Izhoda.
- Na večini sistemov mora biti datoteka ali naprava odprta, da lahko dostopamo do nje.
- Odprta datotečna tabela hrani informacije o vseh datotekah ali napravah, ki so jih odprli procesi.

OSTALO

- Ostale informacije, ki jih lahko OS upravlja, so naslednje:
  - **Identiteta procesa**: Edinstvena identifikacijska številka procesa.
  - **Identiteta starševskega procesa:** Identifikacijska številka starševskega procesa.
  - **Identitete otroških procesov:** Identifikacijske številke vseh otroških procesov, ki jih je ustvaril en starš.
  - **Identiteta gruče procesov:** Številka, ki označuje skupino procesov, ki so med seboj neodvisni ali pa so med njimi starševske vezi.
  - **Identiteta uporabnika:** Identifikacijska številka uporabnika, ki poganja proces.
  - **Identiteta aktivnega uporabnika:** Uporabnik, katerega uporabniške pravice bodo imele efekt med izvajanjem določenega procesa.

- **Identiteta skupine uporabnikov:** Mehanizem kontrole dostopa do sistemskih objektov lahko oblikuje skupine uporabnikov, ki si delijo podobne pravice dostopa. Na takih sistemih lahko proces deluje le z eno ali z več skupinami uporabnikov.

- **Identiteta aktivne skupine uporabnikov:** Prav tako, kot je zaželeno, da se aktivnega uporabnika nadomesti s resničnim, je zaželeno, da se zagotovi mehanizem, ki bo prepoznal aktivno skupino uporabnikov.

- **Identiteta uporabniškega računa:** nekateri sistemi razlikujejo med osebo (uporabnikom), ki ima dovoljenje za uporabo procesa in uporabniškim računom, ki ima dovoljenje za uporabo.

- **Stopnja prioritete:** To je faktor, ki odloča o tem, kam se bo med fazo razvrščanja uvrstil proces. Določa stopnjo pomembnosti in procesi z višjo stopnjo prioritete se izvedejo prvi.

- **Porabljeni čas CPU:** Skupni porabljeni čas procesorja do konca izvedbe procesa

- **Začetni čas:** Realni čas, ko je bil proces ustvarjen in/ali čas, ko je zapustil čakalno vrsto.
- **Začetek izvajanja:** Realni čas, ko naj bi proces začel z izvajanjem. Proces z določenim časom začetka izvajanja mora ostati zadržan do tega časa.
- **Maksimalni čas CPU:** Največja količina procesorskega časa, ki jo proces lahko porabi.
- **Dodelitev pomnilnika:** Skupna količina pomnilnika, ki je dodeljena nekemu procesu.
- **Ukazni niz:** Ime izvajalne datoteke (.exe), ki jo program izvaja in/ali vsi ukazi, ki jih je proces prejel, ko je bil ustvarjen.
- **Terminal:** Na sistemih, kjer uporabniki komunicirajo z sistemom preko terminalskih naprav (monitor/tipkovnica, terminali, telnet povezave, itd.), je prisoten tudi podatek o identifikaciji terminala, ki je v povezavi s procesom.
- **Prekinitveni status:** Vzroki in vrednosti spremenljivk, zaradi katerih je bil proces končan.

Klic nadzornika procesov

- OS mora zagotoviti mehanizem za ustvarjanje procesov.
- Začetne nastavitve za informacije v PCB se lahko razberejo že iz:
  - parametrov Nadzornega klica, ki je ustvaril proces,
  - iz vrednosti starševskega procesa,
  - vrednosti ki pripadajo uporabniku,
  - ali pa vrednosti prebrane iz celotno obremenjenega operacijskega sistema.

Klic nadzornika procesov

- Za začetek mora biti vzpostavljeno delovno stanje strojne opreme za novi proces. Ustvarjena mora biti slika pomnilnika, in podatki za strojne registre, vključujoč programske števce, morajo biti nastavljeni. Pri tem prevladujeta dva osnovna pristopa:
  - klic nadzornika sistema specificira datoteko, ki vključuje program, ki ga bo izvedel novi proces. Operacijski sistem mora naložiti pomnilnik, ki temelji na podatkih shranjenih v datoteki.
  - več formatov izvajalnih datotek(.exe).

Klic nadzornika procesov

- Pri drugem pristopu sta ustvarjenje in izvajanje procesa obravnavani kot dve ločeni funkciji: ***fork*** in ***exec***.

Zahteva ***fork***

- Zahteva *fork* ustvari otroški proces, ki je skorajda identičen klon starševskega procesa; razlikuje se le v enem registrskem ali pomnilniškem naslovu.
- S tem procesoma omogočimo, da se razvejita na različne segmente programske kode.

Zahteva ***exec***

- priredi proces, da izvede program, ki je shranjen v datoteki.
- prepiše trenutno delovno okolje procesa s tistim stanjem, ki je predpisan v datoteki
- S tem, ko otroški proces izvede to nadzorno zahtevo takoj, ko se *fork* konča, **podvojimo storilnost sistema.**
- S tem dovolimo procesu da prepiše sam sebe brez ustvarjanja otroškega procesa, ali pa večkratno procesiranje v istem programu.

Mehanizmi za prekinitev procesa

- Ti mehanizmi vključujejo nadzorne zahteve:
  - ***exit,*** ki prekine zahtevani proces
  - ***abort***, ki dovoli enemu procesu da prekine drugega
  - intervencijo operacijskega sistema, ki proces postavi v neko omejeno aktivnost
  - ***wait*** zahteva omogoči, da se starševski proces začasno prekine, dokler se en ali več otroških procesov ne izvede in zaključi.
- Na nekaterih sistemih, ko je starševski proces končan, so avtomatsko končani tudi vsi otroški procesi. S tem pa bi tudi otroški procesi končali njihove otroške procese. To pa je proces, poznan kot ***kaskadno zaključevanje.***

Naloge klicev

- Kot dodatek, da ustvarjajo in končujejo zahteve, obstajajo klici(calls) tudi zato, da:
  - pošiljajo in kontrolirajo sprejem programskih signalov,
  - za sinhronizacijo hkratnih procesov in
  - za pošiljanje komunikacij med procesi.

KONEC