STEBRI

Temelji vsakega OS-a so štiri funkcionalnosti (rekli smo jim stebri). Katere?

V katero spada RAID.

Upravljanje V/I.

V katero izmed njih spada algoritem Princip ure (angl. Clock Policy)?

Upravljanje pomnilnika.

V katero izmed njih spada Bančniški algoritem?

V razporejanje virov.

V katero izmed njih spada pojem zalepljenost ročice (angl. arm stickiness)? Razložite ta pojem skozi konkreten algoritem.

Upravljanje V/I.

Skeniranje (scan) je naklonjeno tudi nazadnje dospelim zahtevam, ki naslavljajo eno in isto sled (problem monopola oziroma zalepljenosti (arm stickiness).

Pomeni da ročica ostane na isti sledi če tja prihajajo zahteve.

V katero izmed njih spada pojem sočasnosti?

Upravljanje procesov.

OS, MONITORJI, POMNILNIK

a) Pri zgodovinskem pregledu razvoja OS-ov smo pri enostavnih sveženjskih sistemih naleteli na pojem monitor.

Kakšna je bila njegova naloga?

Nadzoruje izvajajoče se aplikacije.

Katere novosti tako prinese monitor kot prvi OS v enostavnih sveženjskih sistemih?

Avtomatizira prej ročno zaporedje za izvajanje ukazov JCL. Zaščiti pomnilnik.

Kakšno nalogo je v monitorju imel nadzorni jezik zahtev (angl. Job Control Language)?

Priskrbuje ukaze monitorju, katere podatke uporabi in kateri prevajalnik.

Na pojem monitor smo naleteli še pri eni drugi snovi tekom semestra (pazite: njegova naloga tam je drugačna). Pri kateri snovi?

Vzajemno izključevanje

Katere so ključne željene lastnosti strojne opreme zaradi uvedbe monitorja?

Zaščita pomnilnika, ura, priviligirani ukazi, prekinitve

Zapišite bistvena problema, ki sta nato pripeljala do uporabe enostavnih sveženjskih sistemov.

• Uporabnik sam razporeja (ročno rezervira) potreben čas

(npr. nx30 min)

• Nameščanje je vključevalo nalaganje prevajalnika, kode

željenega programa, shranjevanje prevedenega programa

ter njegovo nalaganje in povezovanje

Kaj v kontekstu teh sistemov pomeni zaščita pomnilnika?

Ne dovolimo, da programi spreminjajo del

glavnega pomnilnika v katerem se nahaja monitor

Opišite primer, ki pokaže, da je podpora spremembi lokacije procesne slike v glavnem pomnilniku nujna funkcionalnost OS-a.

Pri teh sistemih se prvič srečamo tudi s pojmom rezidenčnost. Kaj to pomeni v tem specifičnem primeru?

Vedno v glavnem pomnilniku in vedno pripravljen za izvajanje.

Deli procese v glavnem pomnilniku.

S katerimi koncepti zagotavljamo podporo vzajemnemu izključevanju znotraj OS-a?

• Semaforji

• Monitorji

• Predajanje sporočil

Razložite povezavo med i) zunanjo drobitvijo glavnega pomnilnika pri dinamičnem razdeljevanju in ii) postopkom zgoščevanja. Narišite tudi primer.

Particije so različnih velikosti; število

particij se spreminja

• Procesu se dodeli natanko toliko

pomnilnika, kot ga zahteva

• čez čas se pojavijo luknje v pomnilniku

– to imenujemo zunanja drobitev

• Zato moramo uporabiti tehniko

zgoščevanja, da vzdržujemo le en

zvezen blok praznega pomnilniškega

prostora (časovno potratno)

Ali pride tukaj do relokacije? Odgovor argumentirajte.

Prikažite ilustrativen primer prevedbe logičnega naslova v fizičnega pri segmentaciji.

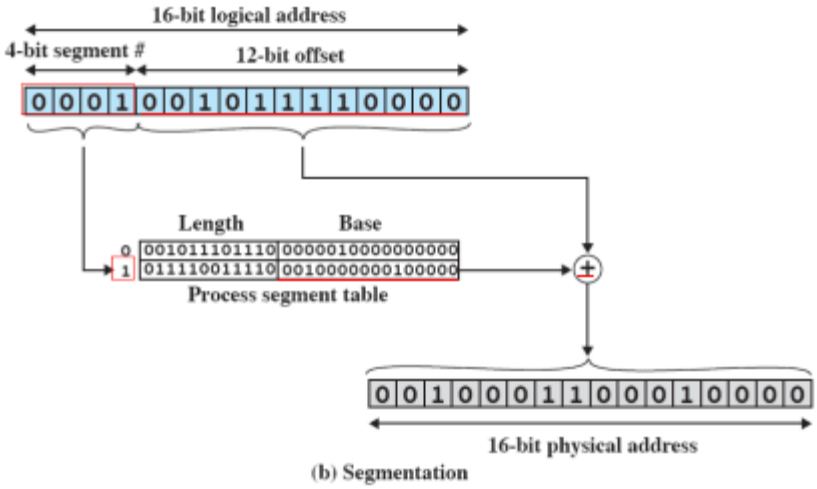


Figure 7.12 Examples of Logical-to-Physical Address Translation

PROCESI, VIRI, NITI

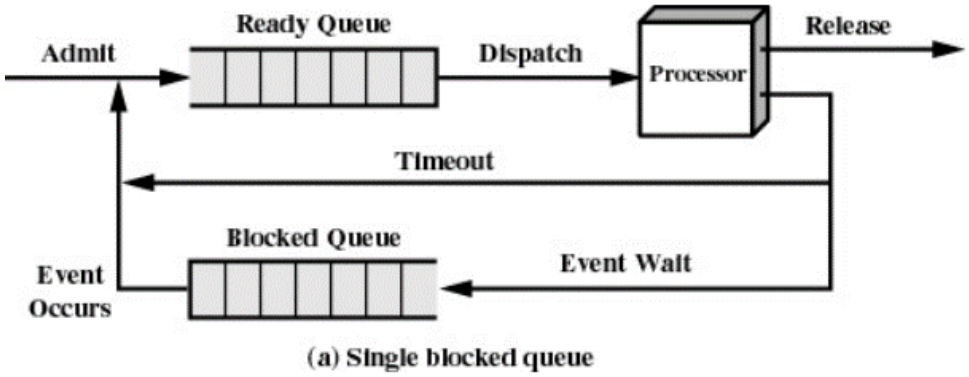
(a) Katere so glavne komponente (na najvišjem nivoju) vsakega procesa?

Program, podatki, PCB.

(b) Katera od teh je implementirana v podatkovni strukturi task struct v Linux-u?

PCB (Process Control Block).

Narišite procesni model petih stanj z uporabo vrst (angl. queues).



(a) Single blocked queue

Kateri stanji še moramo dodati, da dobimo končni procesni model sedmih stanj?

Ready/Suspend, Blocked/Suspend

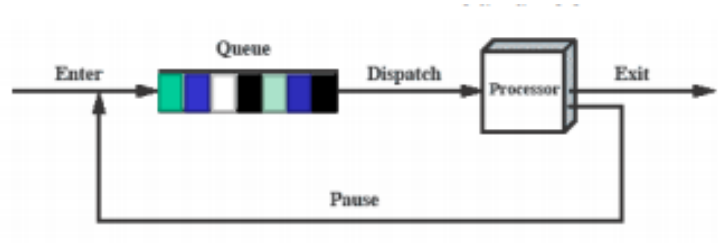
Kakšen problem ima procesni model petih stanj?

Da lahko zmanjka glavnega pomnilnika, saj vsi procesi čakajo na V/I.

Kako ga rešimo?

Dodamo vsaj eno ali v najboljšem primeru dve novi stanji.

Narišite procesni model dveh stanj z vrstami.



Kaj je ključni problem tega modela?

Proces čaka

– Lahko je pripravljen na izvajanje ali pa je:

• Blokiran

– čaka na V/I

• Ne moremo vedno izbrati procesa, ki je

najdlje v vrsti (FIFO), saj je lahko

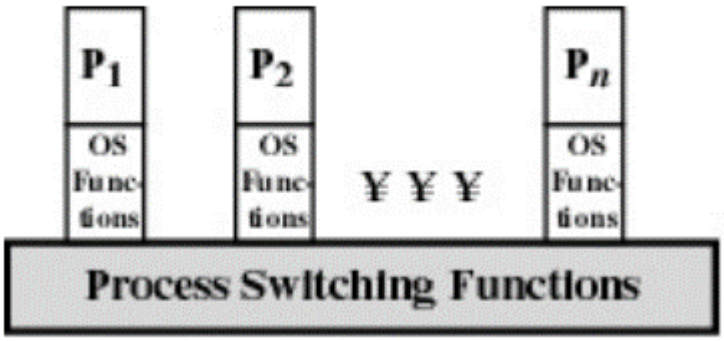
blokiran

Katero stanje je v tem razširjenem modelu povezano s konceptom navideznega pomnilnika?

Blocked queue

Narišite:

izgled izvajanja funkcionalnosti OS-a znotraj uporabniškega procesa ter



izgled procesne slike takšnega procesa.

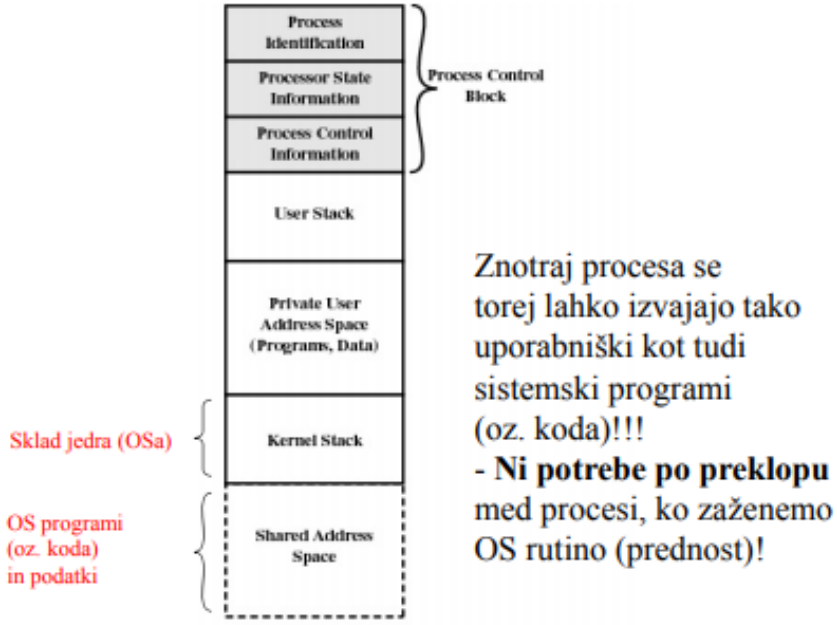


Figure 3.15 Process Image: Operating System Executes Within User Space

– Koda operacijskega sistema se pojavlja v kontekstu

uporabniških procesov

– Proces se izvaja v priviligiranem načinu, ko se izvaja

koda operacijskega sistema

Zapišite ključni dobri lastnosti tako strukturirane procesne slike.

Znotraj procesa se torej lahko izvajajo tako uporabniški kot tudi sistemski programi (oz. koda)!!!

- Ni potrebe po preklopu med procesi, ko zaženemo OS rutino (prednost)!

Katere so glavne skupine informacij v nadzornem bloku procesa (angl. Process Control Block)?

Oznaka procesa(ID), informacije o stanju procesorja, informacije za nadzor procesa

Kakšna je prednost takšne zasnove procesov?

Ni potrebe po preklopu med procesi, ko zaženemo OS rutino.

Kakšne so prednosti niti (v primerjavi s procesi)?

Hitrost ustvarjanja(70x hitreje kot proces),

hitrost zaključevanja,

preklopni čas(context switch pri procesih zelo počasen) in deljenje virov ter medsebojna komunikacija.

Proces lahko ima več niti. Kaj je osnovna enota za i) razporejanje in ii) za lastništvo virov?

Osnovna enota razporejanja je nitka.

Osnovna enota lastništva virov je proces.

Kakšne so prednosti in slabosti uporabe niti na sistemskem nivoju (angl. Kernel Level Threads)?

Prednosti:

-Jedro lahko sočasno razporedi več nitk istega procesa na več procesorjev

-Rutine jedra so lahko tudi same večnitne

-če je blokirana ena nitka procesa, lahko jedro izvaja drugo nitko istega procesa

Slabosti:

-Preklop na drugo nitko istega procesa zahteva preklop v jedrni način

Koncept procesa smo nadgradili s konceptom niti. Katera stanja nitnega modela stanj so še vedno upravljana na nivoju procesa?

Izvajanje, pripravljena, blokirana

(b) Kaj pravi Amdahlov zakon?

Speedup =

$$\frac{time\ t \qquad o\ e}{time\ t \qquad o\ e \qquad xec}$$

$$\frac{1}{(1 \qquad -f)}$$

V povezavi s procesi in nitmi odgovorite: Kaj razporeja razporejevalnik? Kdo si lasti vire?

Razporejevalnik razporeja informacije o virih, ki jih nadzoruje proces, o zgodovini

Koriščenja procesorja in/ali ostalih virov

Vire si lasti proces.

Pravilnost delovanja realno-časovnega OS-a je odvisna od dveh stvari. Katerih?

Pravilnost je odvisna od logičnega rezultata procesiranja in od časa, v katerem je rezultat na voljo.

(b) Katere štiri pristope k razporejenju v realnem času smo spoznali?

Statično tabelarno gnano, statično prioritetno gnano preklopno, dinamično plansko osnovano,

Dinamično celostno(dynamic best effort)

Kako se koncept mikro jedra vidi v arhitekturi MS Windows?

Narejen za enouporabniško večopravilnost. Narišite skico procesne slike v OS-u, kjer je vsaka funkcionalnost OS-a (recimo zahteva za odpiranje datoteke) v glavnem pomnilniku le enkrat.

Kakšna je prednost takšnega načina izvajanja OS-a?

Zapišite definicije:

kritičnega področja

Sekcija kode znotraj procesa, ki potrebuje dostop do deljenega naslovnega prostora, ki ne sme biti dostopana med tem ko nek drug proces dostopa do tega dela kode.

vzajemnega izključevanja.

Kadar je en proces v kritični sekciji izvajanja, ki dostopa do deljenih virov, noben drug proces ne sme dostopati do te kritične sekcije deljenih virov.

smrtnega objema in

Zagotovi, da dva programa ne čakata eden drugega

atomične operacije.

Strojni ukazi, ki niso prekinljivi aka. so neposredni

Opišite, kako deluje algoritem za zaznavo smrtnega objema. Ne pozabite na začetku opisati, kakšen je vhod v algoritem, na koncu pa kakšen je izhod.

1. Označi (mark) vse procese, ki imajo v

vrsticah matrike A le ničle

2. Inicializiraj začasni vektor W = V

3. Najdi takšen indeks i, da proces i ni

označen (unmarked) in da je i-ta vrstica

Q manjša ali enaka W

– Qik Wk za 1 k m.

– če takšne vrstice ni, končaj algoritem

4. če takšno vrstico najdemo

– Označi (mark) proces i in dodaj

korespondenþno vrstico A k W:

Wk = Wk + Aik, for 1 k m

– Vrni se na korak 3

• Smrtni objem obstaja le, če na koncu ostane kakšen proces neoznačen

• Vsak neoznačen proces je v smrtnem objemu

Katere tri mehanizme smo spoznali, ki zagotavljajo podporo vzajemnemu izključevanju znotraj samega OS-a?

Semaforji, monitorji, predajanje sporočil

Katerega od teh mehanizmov smo uporabljali na vajah?

Semaforji.

Katera neprekinljiva strojna ukaza za zagotavljanje vzajemnega izključevanja smo spoznali?

Compare and swap, exchange

Kakšen problem rešujeta?

– Izvedejo se v enem ukaznem ciklu

– Torej niso prekinljivi – so atomični

– Npr. branje in pisanje v enem ciklu,...

Kakšne so njune ključne lastnosti, ki veljajo za oba ukaza?

Uporabno ob poljubnem številu procesov

tako na eno/večprocesorskem sistemu

– So enostavni, zatorej preprosto preverljivi

– Podpirajo veþ kritiþnih obmoþij; vsako ima svojo ključavnico

Kaj je njuna slabost?

Prednosti:

– Uporabno ob poljubnem številu procesov tako na eno/večprocesorskem sistemu

– So enostavni, zatorej preprosto preverljivi

– Podpirajo več kritičnih območij; vsako ima svojo ključavnico

Slabosti:

– čakanje z vrtenjem v prazno (busy wait) troši procesorski čas

– Stradanje je možno, ko proces zapusti kritično območje, več procesov pa čaka na vstop

– Smrtni objem je možen

Narišite in opišite osnovne gradnike grafa alociranja virov.

(a) Resouce is requested (b) Resource is held

Kdaj uporabljamo graf alociranja virov?

V fazi načrtovanja.

OS ima štiri nadzorne podatkovne strukture, ki hranijo informacije o trenutnem stanju vsakega procesa in vira. Katere so te štiri nadzorne strukture?

Pomnilniške tabele, V/I tabele, Datotečne tabele, Procesne tabele

Naštej dogodke, ki se zgodijo ob preklopu med dvema procesoma (angl. context switch)? (Vseh je sedem.)

1. shrani vsebino procesorja (programski števec, …)
2. posodobi nadzorni blok trenutno izvajanega procesa (drugo stanje, …)
3. prestavi nadzorni blok procesa v ustrezno čakalno vrsto glede na novo stanje
4. izberi nov proces za izvajanje
5. posodobi nadzorni blok izbranega procesa
6. posodobi podatkovne strukture upravitelja pomnilnika
7. obnovi vsebino izbranega procesa (na procesor)

Sistem Linux ima v procesnem modelu stanj tudi stanje “zombi”. Zapišite pomen tega stanja.

Procesa ni več, le podatki za starša so še na razpolago.

ALGORITMI

Katere ostale zamenjevalne algoritme smo spoznali (poleg Clock Policy)?

Optimalni, LRU, FIFO, page buffering

Koliko različic algoritma skeniranja smo spoznali?

Krožno (c-scan), pravično (f-scan), navadno (scan), n-step-scan, SSTF (Shortest Seek Time First), FIFO, LIFO

Kako deluje vmesnik/predpomnilnik za strani (angl. Page Buffering), ki smo ga tudi omenili pri zamenjevalnih algoritmih?

– FIFO z izboljšano uþinkovitostjo:

• Stran, ki je bila izbrana za zamenjavo, se raje

doda v enega izmed dveh spiskov (v gl. pomnil.)

• V primeru, da je stran spet potrebna, je še vedno v

glavnem pomnilniku

Kateri algoritmi za kratkoročno razporejanje uporabljajo oceno časa izvajanja procesa za sprejem odločitve?

SPN(Shortest process next) , SRT(Shortest remaining time), HRRN(Highest response ration next)

Kdaj dobimo iz algoritma RR algoritem FIFO?

Če razporejamo brez quantoma (q).

METODE ZASEGANJA

Pri upravljanju pomožnega pomnilnika je zelo pomemben postopek zaseganja prostora za datoteke. Katere tri metode zaseganja smo spoznali?

Zvezno, verižno, indeksirano.

(b) Kako izgledajo tabele FAT pri vsaki izmed njih?

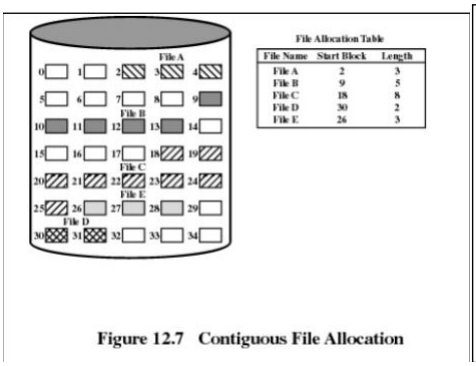


Figure 12.7 Contiguous File Allocation

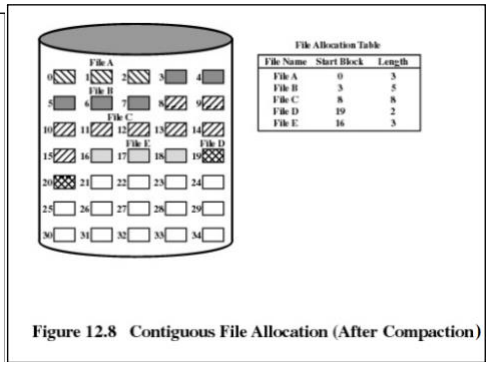


Figure 12.8 Contiguous File Allocation (After Compaction)

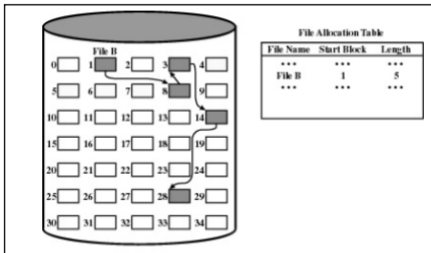


Figure 12.9 Chained Allocation

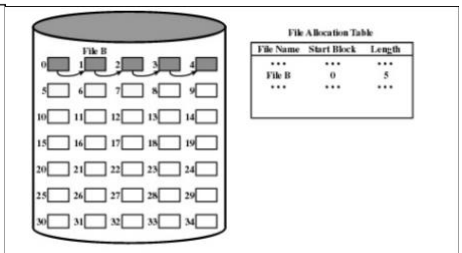


Figure 12.10 Chained Allocation (after consolidation)

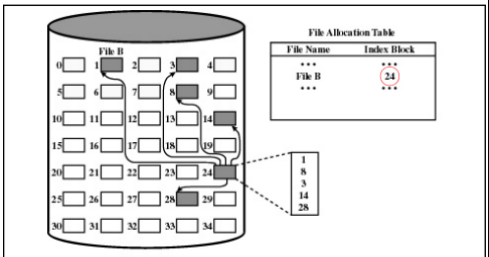


Figure 12.11 Indexed Allocation with Block Portions

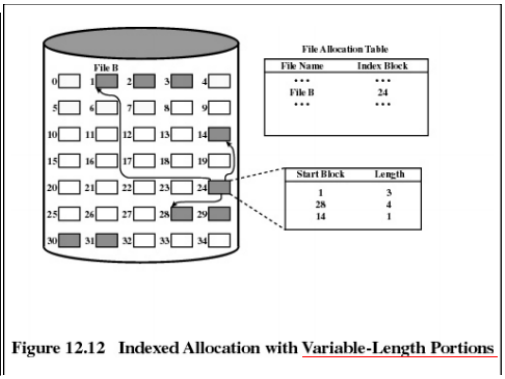


Figure 12.12 Indexed Allocation with Variable-Length Portions

BASH

V Bash-u zapišite funkcijo z imenom ostanek, ki vrne ostanek po deljenju dveh vhodnih argumentov (prvega deliš z drugim), katera podamo ob klicu funkcije.

```
ostanek (){

prva=$1

druga=$2

vsota=$2

while [ $vsota -le $prva ]

do

vsota=$((vsota+druga))

done
```

echo $((prva-(vsota-druga)))

}

Povratni sklici ne spadajo v regularne jezike, a so vseeno pogosto na voljo v kombinaciji z regularnimi izrazi. Kaj predstavlja naslednji izraz: ([01]+)\1 ?

Dvakratne vzorce binarnih števil

Zapišite primer absolutne in relativne poti do procesne slike procesa s PID-om 4433. Trenutno se nahajate v mapi /home/admin.

Abs : /proc/4433

Relative : ../../proc/4433

Zapišite štiri ukaze, ki delajo s procesnimi slikami.

Ps, pidof, pstree, top

Napišite kodo, ki prestreže signal INT in pokliče funkcijo rokovalnik.

trap rokovalnik SIGINT

Napišite kodo, ki ignorira signal INT.

trap '' SIGINT

Kaj naredi ukaz man grep | grep \<the\> ?

Označi besedo "the" če stoji sama.

Kako bi pogledali datotečne deskriptorje procesa cat?

ls /proc/$(pidof cat)/fd/

Zapišite štiri ukaze, ki delajo z uporabniki.

useradd, usermod, who, whoami

Kako bi s pomočjo preusmerjenja zapisali v datoteko test.txt besedilo test?

Ustvarite trdo povezavo z imenom test2.txt na datoteko test.txt.

ln test.txt test2.txt

Kako bi preverili koliko trdih povezav ima datoteka test.txt?

ls -l

Kaj naredi ukaz chmod 1752 test.txt? Ali se spremeni tudi datoteka test2.txt, ki je trda povezava na datoteko test.txt?

Datoteki test.txt doda sticky bit, uporabniku da vse pravice (read, write, execute), skupinam da execute in read pravice, ostalim samo write pravice. Da.

Kako dobimo pomoč za vgrajene ukaze v Linux-u? Kako preverimo, če je ukaz vgrajen?

Help. Type.

Pognali smo ls -l in med drugim dobili sledeči izpis:

drwx-wx--T 46 student devs 4096 jun 17 14:32 shared

Pravice nad imenikom iz zgornjega izpisa zapišite v osmiškem zapisu.

1730

V katerih enotah je podana številka 4096? Bodite natančni!

Bytih

Kako izpišemo absolutno pot imenika shared?

realpath shared

Koliko podimenikov je v imeniku shared?

ls -l shared | grep -c ^d

Ali lahko uporabnik ana, ki ni v skupini devs in nima pravic izvajati kot uporabnik z UID 0, izbriše zgornji imenik? Zakaj?

Pognali smo ls -la in med drugim dobili sledeči izpis:

-rw-r----- 2 student student 31 Jun 20 18:05 .profile

Zapišite velikost datoteke v bajtih.

31

Kaj pomeni . v .profile?

Da je datoteka skrita.

Koliko mehkih povezav kaže na to datoteko glede na podani izpis? Utemeljite!

Iz tega zapisa ne moremo izvedeti.

Kaj nam pove prvi - v zgornjem zapisu? Bodite natančni!

Kakšnega tipa je element.

Kako bi pogledali, koliko prostora zavzame zgornja datoteka na disku?

Du profile

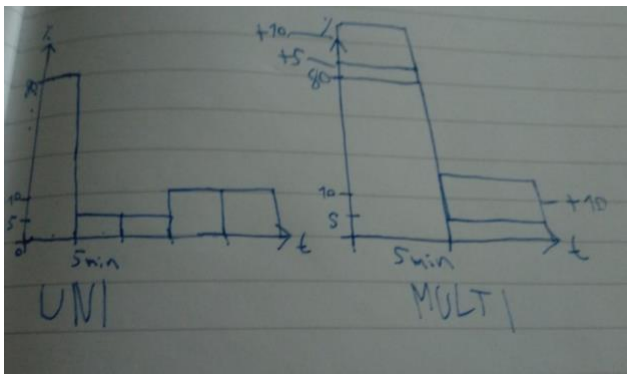
MULTIPROGRAMIRANJE-UNIPROGRAMIRANJE

Izračunajte učinek multiprogramiranja v primerjavi z uniprogramiranjem za naslednji (6) primer:

	zahteva 1	zahteva 2	Zahteva 3
procesorske zahteve	80%	5%	10%
trajanje	5 min	10 min	10 min
pomnilniške zahteve	100 M	50 M	25 M
potrebuje disk?	ne	ne	da
potrebuje terminal?	ne	da	ne
potrebuje tiskalnik?	ne	ne	da

Na voljo imamo 200 M pomnilnika. Izračunati morate: uporabo procesorja v %, uporabo pomnilnika v %, uporabo diska v %, uporabo tiskalnika v %, čas celotnega izvajanja v minutah ter zmogljivost v zahtevah/uro, in sicer tako za primer uniprogramiranja, kot za primer multiprogramiranja. Pomagate si lahko z utilizacijskim histogramom za vsak vir.

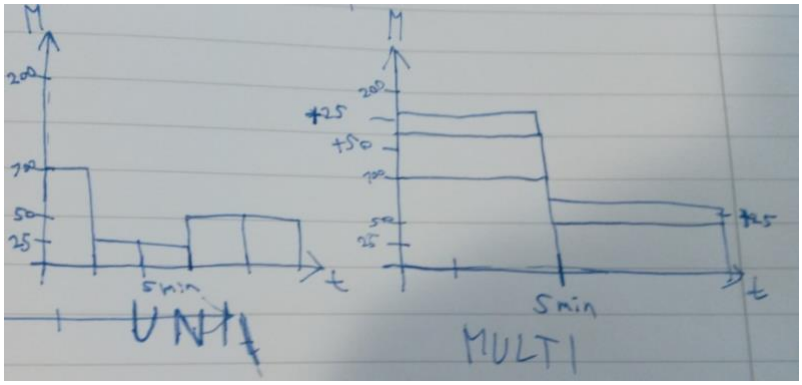
Razdeliš čas na iste časovne enote in jih po procesorskih zahetvah razdeliš v graf.



500 dobiš največji delež * število intervalov v grafu.

UNI: (80+5+5+10+10) / 500 * 100 = 22%

MULTI: (80+5+5+10+10) / 200 * 100 = 55%

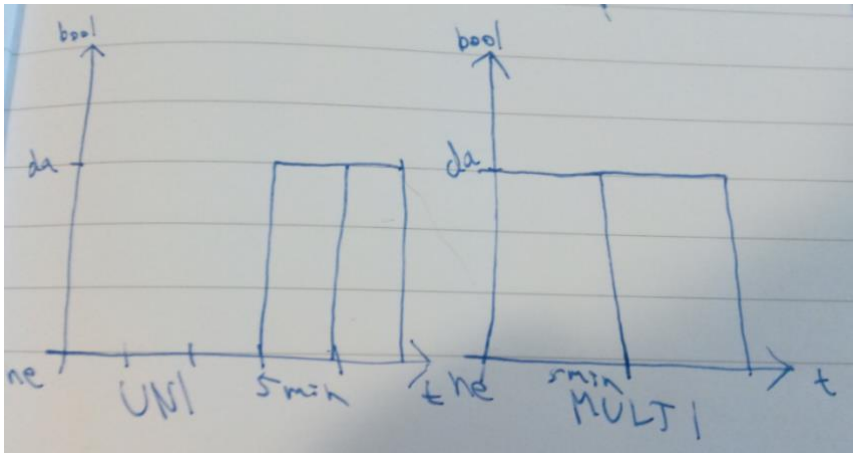


200*5 = 1,000

UNI: (100+25+25+50+50) / 1000 * 100 = 25%

MULTI: (100+25+25+50+50) / 400 * 100 = 62.5%

Za disk in tiskalnik sta enaka rezultata in grafa.



UNI: 2/5 * 100 = 40%

MULTI: 2/2 * 100 = 100%

Za terminal je rezultat enak, graf je malo drugačen.

Kateri dodatni strojni značilnosti smo potrebovali, da smo lahko implementirali podporo multiprogramiranju?

V/I prekinitve in DMA

Sveženjsko multiprogramiranje in delitev procesorskega časa uporabljata multiprogramiranje, a z različnim ciljem. Kaj je glavni cilj sveženjskega multiprogramiranja in kaj delitve procesorskega časa?

	Sveženjsko multiprogramiranje	**Delitev procesorskega časa**
Glavni cilj	Maksimizirati uporabo procesorja	Minimizirati odzivni čas
Vir ukazov oprecijskemu sistemu	JCL ukazi, ki spremljajo zahtevo	Ukazi vnešeni preko terminala

Kako se imenuje koncept, ki nadgrajuje enostaven sveženjski sistem v smeri maksimizacije uporabe procesorja?

Multiprogramiranje.

Kako pa koncept, ki želi minimizirati odzivni čas?

Delitev procesorskega časa.

Imamo proces. Branje zapisa iz diska traja 15 µs. Enako velja za zapisovanje. Izvedba 100 ukazov pa traja le 1 µs.

Kakšna je učinkovitost uniprogramiranja v tem primeru?
15us + 15us + 1us = 31us
1/31 = 0.0323 *100 = 3.23 %
Zakaj je uniprogramiranje v takšnem primeru neučinkovito?
Procesor mora počakati na zaključek V/I ukaza in šele nato lahko nadaljuje z delom.
Kako rešimo to zagato v modernih operacijskih sistemih?
Multiprogramiranje.