**#BEGIN**

**Size**

**Name**

**#PROGRAM**

**#END**

**Pakeisti VM\_INTERUPTED resusrso pranšimus**

**TAI KO GALBUT REIKĖS JEIGU BUS BLOGAI**

**INTERFACE**

-> blokojasi laukadamas IVEDIMO\_ĮRENGINYS resurso

-> blokuojasi laukdamas OPERATYVIOJI\_ATMINTIS resurso (reikia 1 lasteles kad procesorius galėtų vykdyti SYxyz komandą)

-> nustatomį kanalų įrenginio registrai ir vykdoma komanda XCHNG (įvedamas sisteminio pertraukimo numeris)

-> atlaisvinamas resursas ĮVEDIMO\_ĮRENGINYS

-> nustatomas SI registras įvesta reikšme

-> atlaisvinamas OPERATYVIOJI\_ATMINTIS resursas

Naikinti\_Resursa( RS ){

i = RVV( RS ) #Nustatomas resurso VV pagal IV

R = Pašalinti( LPS[i] ) #gaunama nuoroda į pašalinamą elementą

While R <> ø {

ST[R.P] = if ST[R.P] = blocked then ready

else readys #Dirbtinai atblokuojamas procesas

Įjungti( PPS, R.P )

SD[R.P] = PPS #sūnų sąrašas

R.A = „atsakymas“ # atsakymų laukai P,D,A. Kur D – i dalis, A – atsakymo sritis

R = Pašalinti( LPS[i] )

}

NaikintiRD[i] #Naikinamas resursų deskriptorius

Planuotojas()

}

**INTERRUPT**

->jeigu yra VM\_INTERRUPTED resursas, sunaikinti visus esancius

->AR TI==0 THEN TI registro reikšmė atsatoma

-> PLANUOTOJAS ELSE

-> Ar pertraukimas įvyko INTERFACE procese?

TAIP: CASE: SI ==

:5 -> Sukurti resursa PRANEŠIMAS\_VARTOTOJUI (su pranešimu apie operacinės sitemos darbo pabaigą)

:6 -> Sukurti resursa PRANEŠIMAS\_VARTOTOJUI (su pranešimu apie operacines sistemo leidimą rašyti programą)

DEFAULT:

Sukurti resursa PRANEŠIMAS\_VARTOTOJUI (su pranešimu apie bandymą atlikti jam negalimus veiksmus)

NE: nustatomas pertraukimas

-> nustatomas JOB\_GOVERNOR atsakingas uz VM

-> sustabdomas VIRTUAL MACHINE procesas

-> CASE: SI==

:1 VM\_INTERRUPTED (su pranešimu apie įvedimo prašymą) resurso sukūrimas

:2 VM\_INTERRUPTED (su pranešimu apie išvedimo prašymą) resurso sukūrimas

:3 VM\_INTERRUPTED (su pranešimu apie atminties prašymą) resurso sukūrimas

:4 VM\_INTERRUPTED (su pranešimu apie sėkmingą darbo pabaigą) resurso sukūrimas

-> DEFAULT: VM\_INTERRUPTED (su pranešimu apie negalimus veiksmus) resurso sukūrimas

**INPUT PROGRAM**

Blokuojasi laukdamas IVESK\_PROGRAMĄ resurso

-> blokuojasi laukdamas IŠORINĖ\_ATMINTIS resurso

->blokuojasi laukdamas IŠORINIS\_ATMINTIES\_ĮRENGINYS resurso

->blokojasi laukadamas IVEDIMO\_ĮRENGINYS resurso

->nustatomį kanalų įrenginio registrai ir vykdoma komanda XCHNG (kopijuojama iš įvedimo įrenginio parašyta programa į išorinę atmintį)

->atlaisvinamas resursas ĮVEDIMO\_IRENGINYS

->atlaisvinamas resursas IŠORINIS\_ATMINTIES\_ĮRENGINYS

->sukuriamas resursas INPUT\_PROGRAM\_END

-> sustabdomas procesas INPUT PROGRAM

-> sunaikinamas resursas INPUT\_PROGRAM\_END

->sukuriamas resursas PROGRAMA\_IŠORINĖJE\_ATMINTYJE

INPUT

Blokuojasi laukdamas INPUT\_DATA resurso

-> blokojasi laukadamas IVEDIMO\_ĮRENGINYS resurso

->nustatomį kanalų įrenginio registrai ir vykdoma komanda XCHNG (kopijuojama iš įvedimo įrenginio į virtualios mašinos atmintį)

-> atlaisvinamas IVEDIMO\_IRENGINYS resursas

-> naikinamas INPUT\_DATA resuras

-> sukuriamas INPUT\_DATA\_END resursas

Output data

Blokuojasi laukdamas OUTPUT\_DATA resurso

-> blokojasi laukadamas IŠVEDIMO\_ĮRENGINYS resurso

->nustatomį kanalų įrenginio registrai ir vykdoma komanda XCHNG (kopijuojama iš virtualios mašinos atminties į išvedimo įrenginįį)

-> atlaisvinamas IŠVEDIMO\_IRENGINYS resursas

-> naikinamas OUTPUT\_DATA resuras

-> sukuriamas OUTPUT\_DATA\_END resursas

OUTPUT TO USER

Blokuojasi laukdamas PRANEŠIMAS\_VARTOTOJUI resurso

-> blokuojasi laukdamas IŠVEDIMO\_IRENGINYS resurso

->nustatomį kanalų įrenginio registrai ir vykdoma komanda XCHNG (isvedamas pranešimas pagal INFO dali iš resursu deskriptoriausį)

-> atlaisvinamas resursas PRANEŠIMAS\_VARTOTOJUI

->Ar pranešta apie sistemos darbo pabaigą? THEN Sukuriamas resursas DARBO\_PABAIGA **else**

-> ar pranešta apie programos rašyma? THEN

->Sukuriamas resursas IVESK\_PROGRAMĄ

->Blokuojasi laukdamas IVESK\_PROGRAMĄ\_END resurso

-> naikinti resursa IVESK\_PROGRAMĄ

-> aktyvuojamas INPUT PROGRAM procesas **else**

->sukurti resursa OUTPUT\_TO\_USER\_END

->stabdyti procesa OUTPUT TO USER

-> sunaikinti resursa OUTPUT\_TO\_USER\_END

LOADER

Blokavimasis laukiant PAKROVIMO\_PAKETAS resurso

-> blokavimasis laukiant IŠORINIS\_ATMINTIES\_ĮRENGINYS resurso

-> nustatomį kanalų įrenginio registrai ir vykdoma komanda XCHNG (kopijuojama iš išorinės atmineties į operatyviąją atmintį)

-> IŠORINIS\_ATMINTIES\_ĮRENGINYS resurso atlaisvinimas

-> PAKROVIMO\_PAKETAS resurso atlaisvinimas

-> resurso LOADER\_END sukūrimas

-> proceso LOADER sustabdymas

-> resurso LOADER\_END sunaikinimas

VM

Procesorius perjungiamas į vartotojo rėžimą

-> VM programos vykdymas

GOVERNOR

Blokavimasis, laukiant OPERATYVIOJI\_ATMINTIS resurso

->Resurso PAKROVIMO\_PAKETAS sukūrimas (pagal resursas UŽDUOTIS\_IŠORINĖJE\_ATMINTYJE )

-> Blokavimasis, laukiant LOADER\_END resurso

-> resurso PAKROVIMO\_PAKETAS sunaikinimas

-> aktyvuoja LOADER procesą

->Sukurti procesą VIRTUAL\_MACHINE

->Blokavimasis, laukiant proceso VM\_INTERRUPTED pranešimo

-> CASE: pagal VM\_INTERUPTED pranešimą

:ar programa prašo įvedimo?

->atlaisvinti VM\_INTERRUPTED

->sukuriamas resursas INPUT\_DATA

-> blokuojasi laukadmas resurso INPUT\_DATA\_END

-> sunaikinamas INPUT\_DATA resursas

->aktyvuojamas INPUT DATA procesas

->aktyvuojamas VIRTUAL MACHINE procesas

->einama į vietą kur laukiamas VM\_INTERUPTED

:ar programa prašo išvedimo?

->atlaisvinti VM\_INTERRUPTED

-> sukuriamas resursas OUTPUT\_DATA

-> blokuojasi laukdamas OUTPUT\_DATA\_END resurso

->sunaikinamas OUTPUT\_DATA resursas

->aktyvuojamas OUTPUT DATA procesas

->aktyvuojamas VIRTUAL MACHINE procesas

->einama į vietą kur laukiamas VM\_INTERUPTED

:ar programa prašo papildomos atminties?

->atlaisvinti VM\_INRRUPTED resursa

->sukurti resursa ASK\_MEMORY

->Blokuojasi laukdamas MEMORY\_GIVEN resurso

-> naikinamas ASK\_MEMORY resursas

-> aktyvuojamas ADDITIONAL MEMORY procesas

-> aktyvuojamas VIRTUAL MACHINE procesas

->einama į vietą kur laukiamas VM\_INTERUPTED

: Ar programa sėkmingai baigė darbą? THEN

->atlaisvinti VM\_INRRUPTED resursa

-> sukurti resursa PRANEŠIMAS\_VARTOTOJUI (su pranešimu apie sėkminga programos darbo pabaiga)

-> blokuojasi laukdamas OUTPUT\_TO\_USER\_END resurso

-> sunaikina OUTPUT\_TO\_USER resursa

-> aktyvuoja OUTPUT\_TO\_USER procesą ELSE

->atlaisvinti VM\_INRRUPTED resursa

-> sukurti resursa PRANEŠIMAS\_VARTOTOJUI (su pranešimu apie nesėkminga programos darbo pabaiga)

-> blokuojasi laukdamas OUTPUT\_TO\_USER\_END resurso

-> sunaikina OUTPUT\_TO\_USER resursa

-> aktyvuoja OUTPUT\_TO\_USER procesą

-> naikinti procesa VIRTUAL MACHINE

-> sukurti fiktyvų resursą UŽDUOTIS\_IŠORINĖJE\_ATMINTYJE

Main

Blokavimasis laukiant UŽDUOTIS\_IŠORINĖJE\_ATMINTYJE

-> Ar resursas fiktyvus? THEN

-> naikinti JOB\_GOVERNOR sukūrusį fiktyvų resursą ELSE

-> sukurti JOB\_GOVERNOR (perduoti resursą UŽDUOTIS\_IŠORINĖJE\_ATMINTYJE)

Additional memory

Blokuojasi laukdamas ASK\_MEMORY resurso

-> ar neviršytas limitas? THEN

-> blokuojasi laukdamas OPERATYVIOJI\_ATMINTIS resurso

-> priskiriama atmintis virtualiai mašinai

-> sukuriamas pranešimas MEMORY\_GIVEN (pranešimas su papildoma informacija apie sėkmingą darbo pabaigą)

-> naikinamas ASK\_MEMORY resursas ELSE

-> sukuriamas resursas MEMORY\_GIVEN (pranešimas su papildoma informacija apie nesėkmingą darbo pabaigą)

-> naikinamas ASK\_MEMORY resursas