
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Operativni sistemi 2 (13E113OS2, 13S113OS2)

Nastavnik: prof. dr Dragan Milićev

Asistent: Živojin Šuštran

Školska godina: 2023/2024. (Zadatak važi počev od januarskog roka 2024.)

Projekat za domaći rad

- Projektni zadatak –

Verzija dokumenta: 1.0

Važne napomene: Pre čitanja ovog teksta, **obavezno** pročitati opšta pravila predmeta i pravila vezana za izradu domaćih zadataka! Pročitati potom ovaj tekst **u celini i pažljivo**, pre započinjanja realizacije ili traženja pomoći. Ukoliko u zadatku nešto nije dovoljno precizno definisano ili su postavljeni kontradiktorni zahtevi, student treba da uvede razumne pretpostavke, da ih temeljno obrazloži i da nastavi da izgrađuje preostali deo svog rešenja na temeljima uvedenih pretpostavki. Zahtevi su namerno nedovoljno detaljni, jer se od studenata očekuje kreativnost i profesionalni pristup u rešavanju praktičnih problema!

Uvod

Cilj ovog zadatka jeste implementacija dela sistema za virtuelnu memoriju. Sistem za virtuelnu memorije treba da obezbedi sledeće funkcionalnosti: zamenu stranica na prostor za zamenu (engl. *swap space*) i sprečavanje pojave zvane *thrashing*. U sistemu može da postoji više procesa koji dele fizičku memoriju. Treba obezbediti mogućnost zamene stranica prebacivanjem stranica na hard disk. Za pristup prostoru za zamenu na hard disku obezbeđene su funkcije za pristup blokovima na disku.

Zadatak se sastoji od dva dela. Prvi deo je obavezan i za uspešnu odbranu projektnog zadatka student mora da ga uradi. Algoritmi za zamenu stranica, za alokaciju slobodnog prostora, deljenje stranica, učitavanje stranica na zahtev i ostali algoritmi i optimizacije se ostavljaju u nadležnosti samog rešenja i biće razmatrani samo u okviru testiranja performansi sistema.

Opšti zahtevi

Odnos projekta i korisničke aplikacije

Tražene podsisteme treba realizovati na jeziku C. Korisničku aplikaciju, koja sadrži test primere, prevesti nezavisno u konzolni program. U datoj aplikaciji biće prisutna i funkcija *main*. Korisnička aplikacija će napraviti određeni broj procesa korišćenjem sistemskog poziva *fork*. Korisnička aplikacija pristupa uslugama operativnog sistema jedino putem sistemskih poziva.

Odnos projekta i ostatka operativnog sistema

Dat je operativni sistem xv6¹ (programski kod se nalazi na sajtu predmeta). Zadatak studenta je da izmeni deo operativnog sistema xv6 tako da podrži zamenu stranica virtuelne memorije čije je opis dat u ovom projektu. Izradom projekta se ni na koji način ne sme ugroziti ispravno funkcionisanje ostalih delova operativnog sistema. Svaki eventualni problem koji se pojavi po pokretanju projekta biće smatran kao greška pri izradi projekta. Deo koda koji je obezbeđen u okviru postavke projekta je pažljivo napisan, i ukoliko se koristi u skladu sa uputstvom za rad, ne može prouzrokovati nikakve probleme i greške pri izvršavanju. Dati kod dozvoljeno je menjati u onoj meri koja je potrebna za realizaciju traženih zahteva, ali nikako uklanjanjem već postojećih funkcionalnosti i menjanjem postojećih sistemskih poziva.

Razvojno okruženje

xv6 operativni sistem se izvršava na emulatoru u okviru operativnog sistema domaćina (Linux x64). Virtuelna mašina sa instaliranim operativnim sistemom domaćinom, emulatorom i svim potrebnim alatima za prevođenje je data na sajtu predmeta. Ista ta virtuelna mašina će biti korišćena i na odbrani projekta. Predviđeno razvojno okruženje je CLion (studenti mogu nabaviti akademsku licencu na sajtu proizvođača sa studentskim email nalogom). Uputstvo za podešavanje projekta i testiranja je dato na sajtu predmeta. S obzirom da se razvija kernel operativnog sistema standardna biblioteka C jezika nije dostupna.

¹ Detaljan opis xv6 operativnog sistema se može naći u knjizi na adresi (pristupljeno: 9.11.2023. godine)
<https://pdos.csail.mit.edu/6.S081/2023/xv6/book-riscv-rev3.pdf>

Prvi deo: Zamena (20 poena)

Uvod

Za evidenciju virtuelnog adresnog prostora jezgro operativnog sistema koristi hardversku podršku za straničnu organizaciju virtuelne memorije. Tabela stranica je organizovana u tri nivoa. Način preslikavanja treba da bude isti kao i u operativnom sistemu xv6. Na početku rada sistemu će biti dodeljen jedan procesor, fizička memorija fiksne veličine i prostor na disku za zamenu fiksne veličine. U sistemu se izvršava više procesa, pa je potrebno obezbediti adekvatnu sinhronizaciju. Veličina virtuelne adrese je 39 bita, veličina stranice je 4KB, dok je veličina bloka na disku 1KB. U tabeli stranica za svaku stranicu postoje biti za prava pristupa, bit zaprljanosti, bit referenciranja i dva bita koje softver može da koristi za svoje potrebe.

Disk za zamenu stranica

Opis datog interfejsa za pristup disku

Za potrebe zamene stranica studentima su date na raspolaganje funkcije pomoću kojih se pristupa disku za zamenu stranica. Operacije koje su obezbeđene su:

- Upis jednog (i samo jednog) bloka na disk; zadaje se redni broj bloka (blokovi su numerisani počev od 0); podaci za upis se čitaju sa zadatog mesta u memoriji (memorija se ne dealocira u funkciji); da li operaciju treba raditi pomoću uposlenog čekanja ili korišćenjem prekida (1 za uposleno čekanje, 0 za korišćenje prekida);
- Čitanje jednog (i samo jednog) bloka na disk; zadaje se redni broj bloka (blokovi su numerisani počev od 0); podaci se upisuju na zadato mesto u memoriji (memorija se ne alocira u funkciji); da li operaciju treba raditi pomoću uposlenog čekanja ili korišćenjem prekida (1 za uposleno čekanje, 0 za korišćenje prekida);

Date operacije se mogu izvršavati u dva režima. Jedan režim je transfer sa uposlenim čekanjem, dok je drugi režim transfer korišćenjem prekida. U oba režima prekidi se dozvoljavaju u toku trajanja operacije prenosa. Prvi režim se koristi kada promena konteksta nije poželjna., npr. prilikom inicijalizacije sistema dok ne postoje korisnički procesi. Prilikom zadavanja operacije disku, tekući proces zaključava bravu koja ograničava pristup disku, dok je oslobađa tokom samog transfera.

Funkcije za pristup disku na jeziku C

Deklaracije datih funkcija date su u zaglavlju `defs.h`, a implementacija je u fajlu `virtio_disk.c`:

```
void write_block(int blkn, uchar* data, int busy_wait);  
void read_block(int blkn, uchar* data, int busy_wait);
```

Algoritam za zamenu stranica

Zamenu stranica treba raditi za stranice korisničkih procesa, dok stranice jezgra operativnog sistema treba da uvek budu u operativnoj memoriji. Dodatno, stranice korisničkih procesa koje su potrebne za obradu stranične greške treba da uvek budu u operativnoj memoriji. Sve

stranice koje se menjaju potrebno je čuvati na disku za zamenu (ne očekuje se da se stranice sa kodom učitavaju sa diska gde se nalazi izvršni fajl procesa).

Zamena stranica se radi prilikom stranične greške ili prilikom proširenja adresnog prostora procesa sistemskim pozivom kada je cela fizička memorija popunjena. Za izbor stranice za zamenu u sistemu koristi se aproksimacija LRU algoritma sa dodatnim bitima istorije referenciranja. Zamena stranica se radi globalno. Za implementaciju algoritma dozvoljeno je korišćenje prostora za smeštanje svih potrebnih struktura podataka u operativnoj memoriji i na disku za zamenu. Taj prostor se smatra da pripada jezgru operativnog sistema.

Perioda na koju se očitavaju biti referenciranja stranice se ostavljaju u nadležnosti samog rešenja.

Drugi deo: Trashing (10 poena)

Uvod

Cilj drugog dela projekta jeste da se obezbedi sprečavanje pojave zvane *trashing*. Rešenje treba da detektuje pojavu zvane *trashing* pomoću praćenja radnog skupa procesa. Kada se detektuje pojava potrebno je neke procese suspendovati, sve dok pojava zvana *trashing* ne prestane. Biranje koji će se proces suspendovati, kao i na koju periodu se očitavaju biti referenciranja se ostavlja u nadležnosti samog rešenja.

Testovi

Javni testovi

Javni test-program služi da pomogne studentima da elementarno testiraju svoj projekat. Ovi testovi neće obavezno pokriti sve funkcionalnosti koje projekat treba da ima, ali će testirati većinu tih funkcionalnosti. Da bi se projekat uopšte odbranio, neophodno je da projekat sa javnim testom radi u potpunosti ispravno. Studentima se preporučuje da pored javnog testa naprave i svoje iscrpne testove koji će im pomoći da što bolje istestiraju svoj projekat. Programi koji dolaze uz xv6 operativni sistem se takođe mogu smatrati kao javni testovi ako mogu da se izvrše u sistemu sa zamenom stranica.

Tajni testovi

Tajni testovi detaljnije testiraju sve zahtevane funkcionalnosti u različitim regularnim i neregularnim situacijama (greške u pozivu ili radu), i nisu unapred dostupni studentima.

Testovi performansi

Testovi performansi mere vreme izvršavanja procesa i efikasnost u korišćenju resursa. Ovi testovi nisu obavezni, i mogu, ali ne moraju, doneti dodatne bodove u predroku posle nastave za do 20 najboljih odbranih radova. Za potrebe povećanja performansi, studentima je dozvoljeno izmena datog koda, kao i optimizacija/modifikacija traženih algoritama. Posebno će biti cenjena ona rešenja gde se optimalno koristi raspoloživi prostor u operativnoj memoriji i na disku (npr. deljenjem stranica, kopiranjem stranica na zahtev, dinamičkim učitavanjem, itd.), kao i ona rešenja koja se mogu izvršavati na više procesora.

Zaključak

Potrebno je realizovati opisane podsisteme prema datim zahtevima na jeziku C. Testiranje se vrši u laboratorijama katedre na računarima pod operativnim sistemom Windows 10 x64. Virtuelna mašina sa sajta predmeta biće dostupna za vreme odbrane.

Pravila za predaju projekta

Projekat se predaje isključivo kao jedna zip arhiva. U arhivu smestiti samo fajlove sa kodom (.c, .cpp, .h i slične) koji su rezultat izrade projekta. Opisani sadržaj ujedno treba da bude i jedini sadržaj arhive. Arhiva ne sme sadržati ni izvršne fajlove, ni biblioteke, ni bilo kakve testove, niti bilo šta što iznad nije opisano, pogotovu ne sme sadržati repozitorijum koda. Projekat je moguće predati više puta, ali do trenutka koji će preko imejl liste biti objavljen za svaki ispitni rok i koji će uvek biti pre ispita, po pravilu prvi radni dan pre ispita. Na serveru uvek ostaje samo poslednja predata verzija i ona će se koristiti na odbrani. Za izlazak na ispit neophodno je predati projekat (prijava ispita i položeni kolokvijumi su takođe preduslovi za izlazak na ispit). Nakon isteka roka za predaju, projektni zadaci se brišu sa servera, pa je u slučaju ponovnog izlaska na ispit potrebno ponovo postaviti ažurnu verziju projektnog zadataka.

Sajt za predaju projekta je https://rti.etf.bg.ac.rs/domaci/index.php?servis=os2_projekat

Zapisnik revizija

Ovaj zapisnik sadrži spisak izmena i dopuna ovog dokumenta po verzijama.

Verzija 1.0

| Strana | Izmena |
|--------|--------|
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |