

Opis

Učitavanje realizovano je u funkciji `void load(string path_to_obj_file, Memory mem)`. Ona prima putanju do objektnog fajla i memoriju kojom upravlja. Iz objektnog fajla, koristeći odgovarajuće klase koje predstavljaju tabelu simbola, sadržaj i relokacioni zapis sekcije, učitava fajl, vrši ispravljanje korišćenjem relokacionih zapisa, i na kraju popuni sadržaj memorije ovim sadržajem.

Emulator je realizovan kao klasa `Context`, koja predstavlja kontekst izvršavanja i nad kojom se izvršava objektni kod.

Klase koje su deo implementacije date su u nastavku.

- `Log` – klasa koja je korišćena za vođenje logova tokom izvršavanja. Sadrži 2 funkcije – `std(string log) i error(string log)`, koje se koriste za standardno logovanje i za logovanje grešaka.
- `ContentTable` – predstavlja sadržaj sekcije. Ima metode za dodavanje vektora bajtova, kao i za ispis na standardni zadati `ostream`, i pretvaranje sadržaja u `string`.
- `RelocationTable` – klasa koja predstavlja tabelu za relokacije. Sadrži svoju strukturu `Entry`, koja predstavlja jedan ulaz u tabelu za relokacije. Sadrži metode za dodavanje novog zapisa o relokaciji, kao i dohvatanje vrednosti odgovarajućih zapisa. Klasa sadrži metode za pretvaranje sadržaja u čitljiv oblik (`string`), kao i za ispis na odgovarajući tok podataka.
- `SymbolTable` – klasa koja predstavlja tabelu simbola. Sadrži svoju strukturu `Entry`, koja predstavlja jedan ulaz u tabelu simbola. Sadrži metode za dodavanje novog ulaza, kao i dohvatanje vrednosti odgovarajućih ulaza. Klasa sadrži metode za pretvaranje sadržaja u čitljiv oblik (`string`), kao i za ispis na odgovarajući tok podataka.
- `Context` - predstavlja kontekst izvršavanja. Sadrži memoriju, registre i flegove za prekide, kao i nit za generisanje prekida sistemskog časovnika. Nad objektom konteksta može se pokrenuti izvršavanje pozivom funkcije `execute`. Ova funkcija prolazi kroz memoriju, šitajući instrukcije i izvršavajući ih, dok se ne generiše prekid sa brojem 0, ili dok se ne pojavigreška.
- `Memory` – klasa koja predstavlja memorijusistema. Adrži metode za dohvatanje i postavljanje bajta, reči i duple reči (1B, 2B i 4B respektivno) sa, odnosno na odgovarajuću adresu.

Prethodno pomenute klase su smeštene fajlove koji sadrže odgovarajući naziv - `<class>.cpp` za implementaciju klase, i `<class>.h` za deklaraciju, gde `<class>` predstavlja puno ili skraćeno ime klase.

Ostali fajlovi koji se koriste su

- `my_util.h` - deklaracije eksterne promeljive tipa klase `Log` koja se koristi za logovanje
- `main2.cpp` – sadrži ulaznu tačku programa, odnosno funkciju `main`, koja prima 1 argument sa komandne linije, a koji predstavlja putnju do fajla sa izvornim kodom i poziva funkciju za asembliranje.

Za prevodjenje je dovoljno pokrenuti make file (make2) koji je dostavljen sa rešenjem, ili prevesti ga ručno pomoću komande (uz prethodno instaliranje verzije 4.9 g++ kompajlera i ncurses biblioteke).

```
g++-4.9 -std=c++0x -o exe2 main2.cpp content_table.cpp line.cpp
log.cpp my_util.cpp reloc_table.cpp sym_table.cpp emulator.cpp
loader.cpp -I. -pthread -lncurses
```

Rešenje je testirano testovima koji su dostavljeni uz rešenje, prostim pokretanjem exe2 fajla sa argumentom naziva testa. Tokom izvršavanja programa na ekranu biće ispisane odgovarajuće poruke koje ukazuju na uspešnost izvršavanja, kao i sve šta program koji se emulira ispisuje na standardni izlaz.

Pokretanjem test1_obj.txt kroz emulator i pritiskom taster 123456789a dobija se izlaz (prikaz izlaza je bez pomoćnih ispisa) [ispis 10 karaktera koje korisnik pritisne u obrnutom poretku]

```
a987654321
```

Pokretanjem test2_obj.txt kroz emulator dobija se izlaz (prikaz izlaza je bez pomoćnih ispisa) [štampanje karaktera abc]

```
abc
```

Pokretanjem test3_obj.txt kroz emulator dobija se izlaz (prikaz izlaza je bez pomoćnih ispisa) [sabiranje niza {1,1,1,1,1,1,1,1,1,2,2,2,2,5,5,10} i ispis zbira na standardni izlaz sa 4 cifre]

```
0040
```

Pokretanjem test4_obj.txt kroz emulator dobija se izlaz (prikaz izlaza je bez pomoćnih ispisa) [ispisuje slovo a periodično sa periodom 0,1s]

```
aaaaaaaaaaa...
```

Pokretanjem test5_obj.txt kroz emulator dobija se izlaz (prikaz izlaza je bez pomoćnih ispisa) [primer poziva f-je sa parametrima – f-ja sabira 2 broja koji su na a steku i vraća rezultat u registru R0. Nakon povratka, ispisuje se rezultat kao i 2 parametra sa kojima je pozvana f-ja]

```
7=5+2
```

Naredbe za instaliranje verzije 4.9 g++ kompajlera date su u nastavku

```
sudo add-apt-repository ppa:ubuntu-toolchain-r/test
sudo apt-get update
sudo apt-get install g++-4.9
```

Naredbe za instaliranje biblioteke `ncurses` date su u nastavku

```
sudo apt-get install libncurses5-dev libncursesw5-dev
```