

Dokumentacija za projekat 31 – Logički

Uvod:

Logički projekat podrazumeva klijent-server arhitekturu gde je klijentima omogućeno razmenjivanje poruka preko servera.

Treba omogućiti pokretanje neograničenog broja klijenata i njihovo opsluživanje od strane servera.

Cilj projekta je definisanje protokola za komunikaciju izmedju komponenti, optimalno rukovanje memorijom, upotreba optimalnih struktura i omogućavanje paralelnog izvršavanja određenih funkcija.

Dizajn:

Unutar servera je implementirana Lista kao glavna struktura za čuvanje informacija o registrovanim klijentima. Sastoji se od dva polja: naziv klijenta i acceptedSocket.

Strukture koje ce služiti za komunikaciju i koriste se i na serveru i na klijentu su:

- REGISTRATION
- MESSAGE_THROUGH_SERVER

Komponent dijagram:



Strukture podataka:

Na serveru je implementirana lista kao struktura podataka koja će služiti za čuvanje konektovanih klijenta, jer pruža mogućnost konektovanja neograničenog broja klijenata, za razliku od niza gde bi morali unapred da definišemo broj klijenata koji mogu da se konektuju.

Sledeće strukture su implementirane radi bolje čitljivosti koda i lakšeg razumevanja šta je poslato/stiglo u buffer.

REGISTRATION - sadrži polja id i name

id - 0 - identifikovaće poruku za registraciju

name - Polje name predstavlja naziv klijenta koji se registruje, i preuzima se unosom sa tastature.

MESSAGE_THROUGH_SERVER - sadrži polja id, name i text

id - 1 - identifikovaće poruku preko servera

id - 2 - zahtev za igrača

id - 3 - zahtev intervala brojeva

id - 4 - greška: igrač je već dodeljen

id - 5 - označava kraj igre

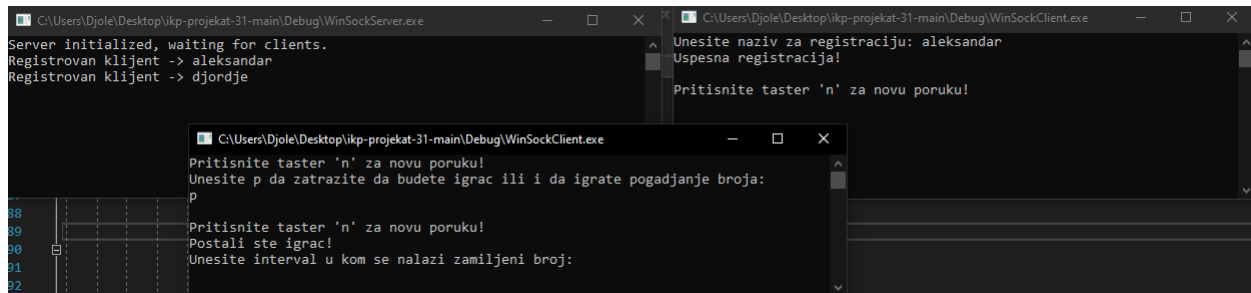
name - naziv klijenta kome se šalje poruka preko servera

text - tekst poruke koja se šalje

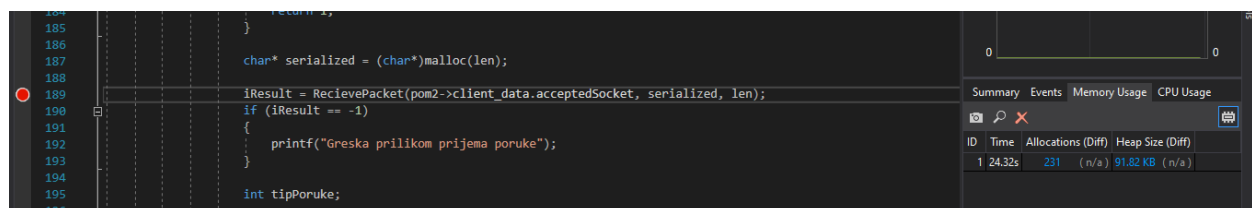
Rezultati testova:

U sledećem primeru je testirano curenje memorije na serveru. Kada pristigne poruka od klijenta, server treba da alocira memoriju odredjeni broj bajtova i da smesti tu pristiglu poruku u alociranu memoriju, zatim da obradi tu poruku, i prosledi je dalje drugom klijentu. Na kraju obrade i slanja poruke drugom klijentu, prethodno zauzeta memorija treba da se oslobodi.

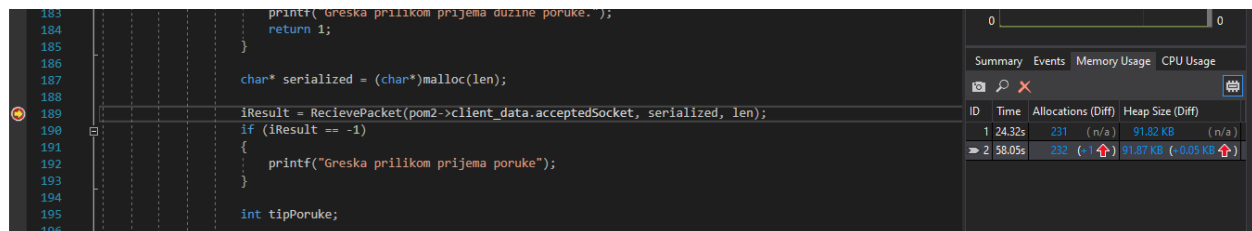
Klijent šalje interval brojeva nekom drugom klijentu, preko servera:



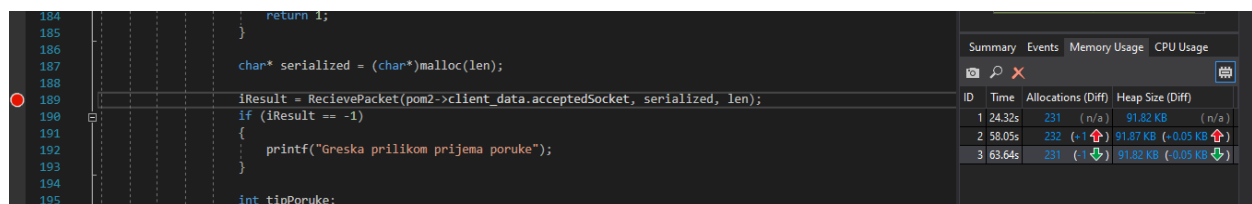
Zauzeta memorija na heap-u servera pre zauzeća memorije za očekivanu poruku:



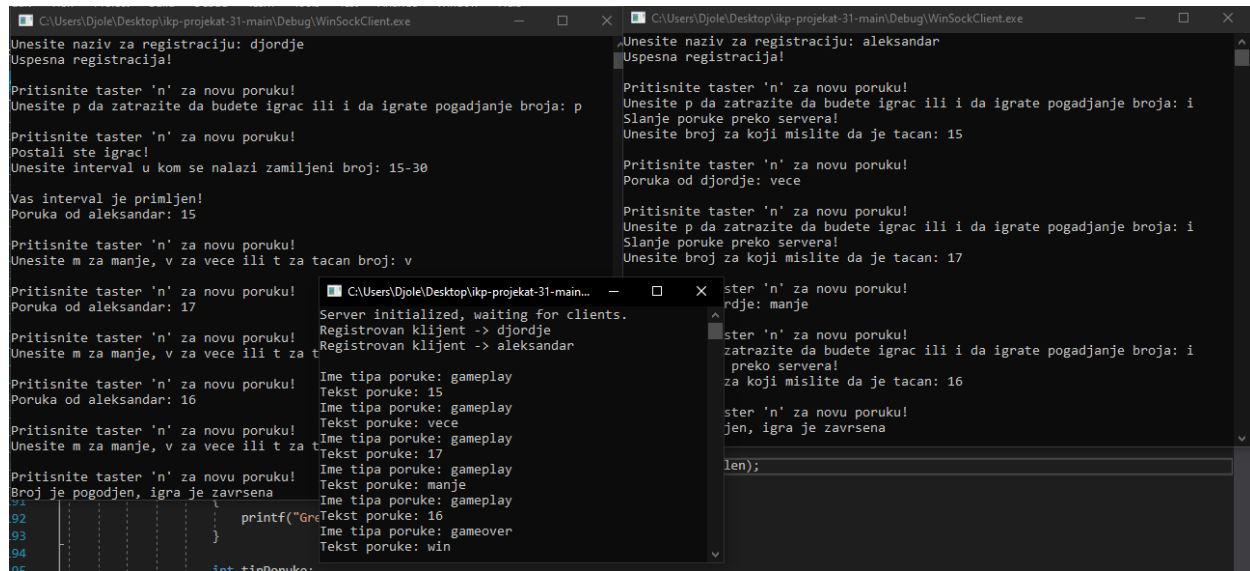
Nakon zauzeća memorije na heap-u servera, vidimo da je zauzeto dodatnih 0.05KB:



Nakon što je server obradio tu pristiglu poruku i prosledio je dalje drugom klijentu, oslobadjamo prethodno alociranu memoriju za poruku i vidimo da je oslobodjeno 0.05KB:



Primer komunikacije klijenata preko servera:



```
Unesite naziv za registraciju: djordje
Uspesna registracija!

Pritisnite taster 'n' za novu poruku!
Unesite p da zatražite da budete igrač ili i da igrate pogadjanje broja: p

Pritisnite taster 'n' za novu poruku!
Postali ste igrač!
Unesite interval u kom se nalazi zamiljeni broj: 15-30

Vas interval je primljen!
Poruka od aleksandar: 15

Pritisnite taster 'n' za novu poruku!
Unesite m za manje, v za vece ili t za tacan broj: v

Pritisnite taster 'n' za novu poruku!
Poruka od aleksandar: 17

Pritisnite taster 'n' za novu poruku!
Unesite m za manje, v za vece ili t za t

Pritisnite taster 'n' za novu poruku!
Ime tipa poruke: gameplay
Tekst poruke: 15

Pritisnite taster 'n' za novu poruku!
Ime tipa poruke: gameplay
Tekst poruke: vece

Pritisnite taster 'n' za novu poruku!
Ime tipa poruke: gameplay
Tekst poruke: 17

Pritisnite taster 'n' za novu poruku!
Ime tipa poruke: gameplay
Tekst poruke: manje

Pritisnite taster 'n' za novu poruku!
Ime tipa poruke: gameover
Tekst poruke: win

92
93
94
95
```

```
Unesite naziv za registraciju: aleksandar
Uspesna registracija!

Pritisnite taster 'n' za novu poruku!
Unesite p da zatražite da budete igrač ili i da igrate pogadjanje broja: i
Slanje poruke preko servera!
Unesite broj za koji mislite da je tacan: 15

Pritisnite taster 'n' za novu poruku!
Poruka od djordje: vece

Pritisnite taster 'n' za novu poruku!
Unesite p da zatražite da budete igrač ili i da igrate pogadjanje broja: i
Slanje poruke preko servera!
Unesite broj za koji mislite da je tacan: 17

Pritisnite taster 'n' za novu poruku!
Poruka od aleksandar: 16

Pritisnite taster 'n' za novu poruku!
Unesite m za manje, v za vece ili t za t

Pritisnite taster 'n' za novu poruku!
Ime tipa poruke: gameplay
Tekst poruke: 16

Pritisnite taster 'n' za novu poruku!
Ime tipa poruke: gameover
Tekst poruke: win
```

```
Server initialized, waiting for clients.
Registрован klijent -> djordje
Registрован klijent -> aleksandar
```

Zaključak:

Na osnovu rezultata testova vidimo da server pravilno rukuje memorijom i da nema curenja memorije unutar serverske aplikacije, na isti način je implementirano i rukovanje memorijom za klijentsku aplikaciju.

Potencijalno unapredjenje:

Na serveru je mogla biti implementirana HashMapa umesto Liste jer predstavlja optimalniju strukturu za pretragu, jer pretražuje po ključu za razliku od Liste koja pretražuje linearno, tj dok se ne dodje do željenog elementa.