

Memory Management

Razvoj Heap Manager-a sa podrškom za više niti, algoritam za traženje slobodnog prostora i praćenje performansi

Potrebno je implementirati **Heap Manager** koji podržava dve osnovne operacije alokacije i dealokacije memorije. Memorija treba da bude organizovana u segmente predefinisane veličine, a broj segmenata treba da zavisi od potreba sistema. Sistem treba da oslobađa neiskorišćene segmente kada ih bude više od 5. Heap manager treba da omogućava izbor između sledećih algoritama za traženje slobodnog prostora:

- **Next fit** algoritam za traženje prvog slobodnog segmenta odgovarajuće veličine.

Funkcionalnosti i zahtevi sistema:

1. Operacije alokacije i dealokacije memorije:

- **void* allocate_memory(int size);**
Funkcija koja alokira memorijski blok od specifične veličine. Alokacija treba da koristi **first fit** algoritam za traženje slobodnog prostora.
- **void free_memory(void* address);**
Funkcija koja oslobađa memorijski blok na datoj adresi. Oslobađanje memorije treba da bude praćeno kontrolom broja slobodnih segmenata, a ako ih ima više od 5, treba ih osloboditi kako bi se smanjila potrošnja memorije.

2. Organizacija slobodne memorije:

- Slobodna memorija treba da bude organizovana u segmente predefinisane veličine. Broj segmenata treba biti dinamičan i zavisi od potreba sistema.
- Granica za oslobađanje neiskorišćenih segmenata je 5 slobodnih segmenata. Kada slobodnih segmenata bude više od 5, sistem treba da oslobodi višak segmenata.

3. Thread safety:

- Heap manager mora biti dizajniran za rad u okruženju sa više niti, koristeći odgovarajuće mehanizme za sinhronizaciju (kao što su mutex-i ili semafori) kako bi se obezbedila bezbednost pristupa memoriji u paralelnim operacijama.

4. Izbegavanje zagušenja u višedretvenom okruženju:

- Implementirati strategiju za izbegavanje zagušenja u okruženju sa više niti, kako bi se osigurala optimalne performanse i efikasno korišćenje memorije.

5. Testiranje u serverskoj aplikaciji:

- Implementirati jednostavnu serversku aplikaciju koja koristi Heap Manager za alokaciju memorije. Server treba da prima zahteve od klijenata i koristi alokaciju memorije za obradu podataka, uz praćenje performansi i upravljanje memorijom.

