

Structuri de Date

Tema 1

Autori:

Alexandra Ștefania Ghiță
David Iancu

Data publicării: 13.03

Data predării: 27.03

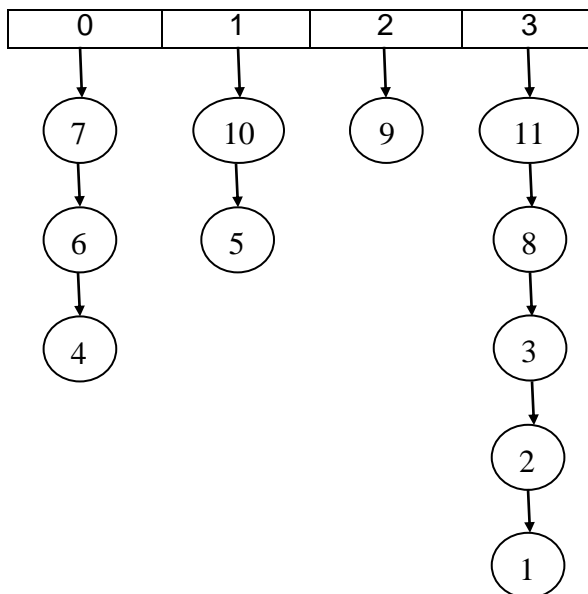
1. Descrierea problemei

Un oficiu de relații cu clienții vrea să își îmbunătățească modul de funcționare folosind un sistem de organizare a oamenilor la ghișee. Oficiul are mai multe ghișee deschise, fiecare fiind specializat pe o anumită problemă, dar putând rezolva și alt tip de probleme. Fiecare ghișeu poate avea în așteptare mai mulți oameni, care sunt ordonați în funcție de prioritate. Dacă un om se așază la un ghișeu și are cea mai mare prioritate, acesta este abordat primul. Noi ghișee pot fi deschise sau unele deja existente pot fi închise, caz în care oamenii sunt redistribuiți pe cât posibil în mod egal. Scopul temei este să realizați un program care să simuleze sistemul cerut.

2. Detalii de implementare

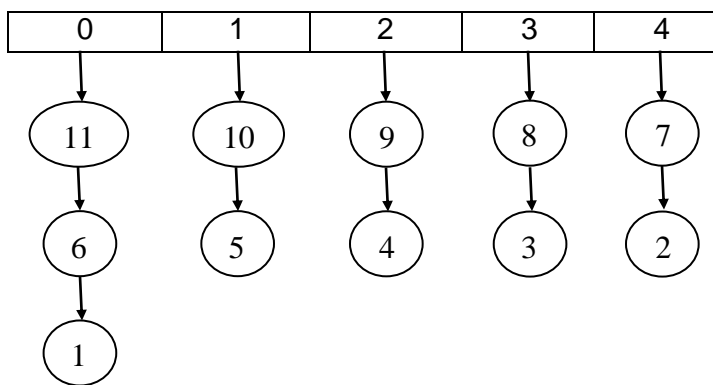
În fiecare moment vor fi N ghișee deschise ($N \geq 1$). Ghișeele vor fi reprezentate sub forma unui vector de lungime N alocat dinamic. Fiecărui ghișeu îi va fi asociată o listă dublu înălțuită care reprezintă oamenii aflați în așteptare la ghișeul respectiv și care va fi ordonată descrescător în funcție de prioritate. Dacă există mai mulți oameni cu aceeași prioritate, aceștia vor fi abordați în ordinea alfabetică a numelui. În cazul în care numărul de ghișee N se modifică, toți oamenii vor fi aranjați într-o nouă listă ordonată descrescător în funcție de prioritate, respectiv alfabetic în funcție de nume, și apoi vor fi împărțiți în mod egal la fiecare ghișeu, așa cum se vede și în exemplul următor. Să presupunem că avem inițial 4 ghișee deschise (0-3), și 11 oameni cu prioritățile de la 1 la 11, așezați ca în imaginea de mai jos:

Vector ghișee:



Ghișeul 0 are 3 oameni în listă, ghișeul 1 are 2, ghișeul 2 are un singur om, iar ghișeul 3 are 5 oameni la rând. Dacă numărul de ghișee deschise se modifică și devine 5, atunci oamenii vor fi mai întâi ordonați în funcție de priorități, și redistribuiți la noul număr de ghișee, astfel încât oamenii cu prioritățile cele mai mari să fie primii în listă, așa cum se poate vedea și în figura de mai jos:

Vector ghișee:



3. Funcții necesare

Funcțiile pe care trebuie să le implementați sunt:

- **deschide_ghisee(N)** - inițializează un număr de N ghișee, fiecare ghișeu având asociată o listă vidă; dacă există deja un număr de ghișee deschise, atunci se redistribuie persoanele existente la noul număr de ghișee. Această funcție precizează câte ghișee sunt deschise, tratând ambele cazuri în care se deschid noi ghișee sau se închid unele deja existente. Funcția nu întoarce nicio valoare.
- **adauga_persoana(nume_persoana, prioritate_persoana, numar_ghiseu)** - adaugă persoana cu numele și prioritatea dată în lista asociată ghișeului cu numărul *numar_ghiseu*. În cazul în care există mai multe persoane cu aceeași prioritate, atunci acestea vor fi ordonate crescător în funcție de nume. Funcția nu întoarce nicio valoare.
- **sterge_persoana(nume_persoana, prioritate_persoana, numar_ghiseu)** - șterge persoana cu numele și prioritatea dată din lista asociată ghișeului cu numărul *numar_ghiseu*. Funcția nu întoarce nicio valoare.
- **sterge_primii_oameni()** - elimină de la fiecare ghișeu primul om din listă (cel cu prioritatea cea mai mare). Funcția nu întoarce nicio valoare.
- **inchide_ghisee()** – eliberează toată memoria alocată (ghișee + listele de oameni). Funcția nu întoarce nicio valoare.
- **calculeaza_numar_total_oameni()** - întoarce numărul de oameni care stau la rând la toate ghișeele. Funcția întoarce un număr întreg.
- **calculeaza_numar_oameni_ghiseu(numar_ghiseu)** - întoarce numărul de oameni care stau la rând la ghișeul cu numărul *numar_ghiseu*. Funcția întoarce un număr întreg.

- **gaseste_pozitie_persoana(nume_persoana, prioritate_persoana, numar_ghiseu)** - întoarce poziția persoanei cu numele și prioritatea dată în rândul asociat ghișeului cu numărul *numar_ghiseu*. Întoarce un număr întreg. Prima poziție din listă este poziția 0, iar în cazul în care omul nu se găsește la rând se întoarce -1.
- **gaseste_persoana_din_fata(nume_persoana, prioritate_persoana, numar_ghiseu)** - întoarce numele persoanei care se află în fața persoanei cu numele și prioritatea dată în rândul asociat ghișeului cu numărul *numar_ghiseu*. Întoarce un vector de tipul char. Dacă persoana nu există în listă sau nu are o altă persoană în față se întoarce "Nu exista."
- **afiseaza_rand_ghiseu(numar_ghiseu)** - afișează numele și prioritatea persoanelor care stau la rând la ghișeul cu numărul *numar_ghiseu*. Formatul de afișare va fi:
`nume_persoana1-prioritate_persoana1;nume_persoana2-prioritate_persoana2;`
 Dacă nicio persoană nu se află la ghișeul dat se va afișa: "*Ghiseul x nu are nicio persoana in asteptare.*", unde x este numărul ghișeului. Afișarea se va face în fișierul primit ca parametru.

4. Rulare

Programul este testat folosind script-ul din cadrul arhivei: ./run.sh

Pentru a testa eliberarea memoriei se poate folosi "make runValgrind".

5. Notare

- 85 puncte obținute pe testele din schelet
- 5 puncte obținute pentru eliberarea integrală a memoriei (punctele vor fi acordate la corectare)
- 10 puncte: README, coding style, codul trebuie să fie comentat, consistent și ușor de citit. De exemplu, tema nu trebuie să conțină:
 - warninguri la compilare
 - linii mai lungi de 80 de caractere
 - tab-uri amestecate cu spații; denumire neadecvată a funcțiilor sau a variabilelor
 - alte situații nespecificate aici, dar considerate inadecvate
- Temele care nu compilează vor primi punctaj 0.

6. Precizări generale

Tema va fi încărcată pe cs.curs.pub.ro sub forma unei arhive zip cu numele Tema1_Grupa_Nume_Prenume.zip (exemplu: Tema1_316CD_Ionescu_Ion_Ionut.zip). Tema este individuală. Colaborarea, respectiv copierea codului de pe internet pentru liste, va fi sancționată ca atare. În cazul în care vor fi prinși studenți cu teme copiate, vor fi sancționați amândoi, indiferent cine a fost sursa. Nu dați temele altor colegi. O temă făcută individual vă va fi de folos mai târziu.

Modificările pe care trebuie să le faceți vor fi făcute doar în fișierul "List.h".

Orice nelămuriri în legătură cu această temă vor fi adresate pe forum.