

Laborator 12 - Tabele de dispersie

Mihai Nan

Mai 2017

1 Probleme de laborator

Observatie

În arhiva laboratorului, găsiți un schelet de cod de la care puteți porni implementarea funcțiilor propuse, având posibilitatea de a le testa funcționalitatea.

1.1 Problema 1 - 1 punct

Analizați funcția `hash` pentru stringuri definită în scheletul de cod oferit în cadrul arhivei de laborator. Pentru implementarea tabelului de dispersie veți folosi fișierele furnizate **`vector.h`** și **`list.h`**.

- **`list.h`** - implementează o listă simplu înlănțuită de șiruri de caractere.

Funcțiile pe care va veni baza sunt **`list_prepend`** - implementată și **`list_delete`** - trebuie să o implementați.

- **`vector.h`** - implementează un vector de liste. Pe acesta îl veți folosi pentru a implementa tabela de dispersie.

Funcțiile de bază sunt **`vec_init`**, **`vec_add`**, **`vec_at`** și **`vec_set`** care vă vor permite să creați și să inițializați un vector pentru a memora tabela de dispersie.

1.2 Problema 2 - 4 puncte

Implementați un hash-table care atunci când găsește 2 elemente cu aceeași cheie să folosească metoda chaining, adică să fie implementat ca un vector de liste înlănțuite și de fiecare dată când apare o valoare cu un anumit hash să o pună în lista corespunzătoare.

1.3 Problema 3 - 4 puncte

Implementați un hash-table care atunci când se generează o cheie care mai există va aplica rehashing (aplicați o funcție hash h1 pentru numere întregi asupra hashului ș.a.m.d.). Faceți acest lucru până când ajungeți la o poziție liberă. Funcția h1 ar putea fi:

```
hash(i)=i*2654435761 mod 2^32
```

Pentru fiecare din problemele 2 și 3 va trebui să definiți o structură care să conțină dimensiunea hashului și elementele propriu-zise. Pentru problema 2 elementele vor fi reținute într-un array de liste ale căror noduri conțin stringuri. Iar, pentru problema 3, elementele vor fi reținute într-un array de stringuri.

Setați dimensiunea hashului 2500.

Testați cele 3 probleme adăugând stringurile din fișierul **`words.txt`** și apoi căutându-le pe unele dintre ele și pe altele care nu fuseseră introduse.

1.4 Problema 4 - 1 punct

Măsurați timpul cât durează căutarea unei valori folosind cele 2 variante de hashtable. Faceți măsurarea și pe o valoare care există, și pe una care nu există în hash table. Trimiteți ca argument în linia de comandă cuvântul pe care îl căutați și măsurați folosind comanda `time`.