

UNIVERSITATEA DE STAT DIN MOLDOVA
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ
SPECIALITATEA INFORMATICA

RAPORT

la disciplina „Algoritme, Structuri de Date și Complexitate”

Lucrarea de laborator nr. 1: Metodele de căutare

Conducătorul științific: Novac Ludmila, dr. conf. univ.

Autor: Cemîrtan Cristian, student din grupa I2101

CHIȘINĂU – 2023

Cuprins

Tema bazei de date alese	3
Rularea metodelor de căutare	4
Metoda secvențială de căutare.....	4
Metoda de căutare în tabelele neordonate structurate arborescent.....	4
Metoda binară de căutare.	5
Metoda de căutare prin tabela de repartizare.	6
Metoda de căutare prin interpolare.....	6
Metoda Fibonacci de căutare.....	7
Lungimile medii de căutare teoretice pentru fiecare metodă de căutare.....	8
Concluzii	9
Anexe	10

Tema bazei de date alese

Tema aleasă pentru baza de date, experimentată în lucrarea de laborator realizată, este „Clienții internet”. Baza de date cuprinde în sine un singur tabel cu următoarele atribute: (id, firstName, secondName, email, female, tax, ip, country), cu tipurile de date:

- id – size_t
- firstName – char[50];
- secondName – char[50];
- email – char[100];
- female – bool;
- tax – double;
- ip – char[4];
- country – char[3].

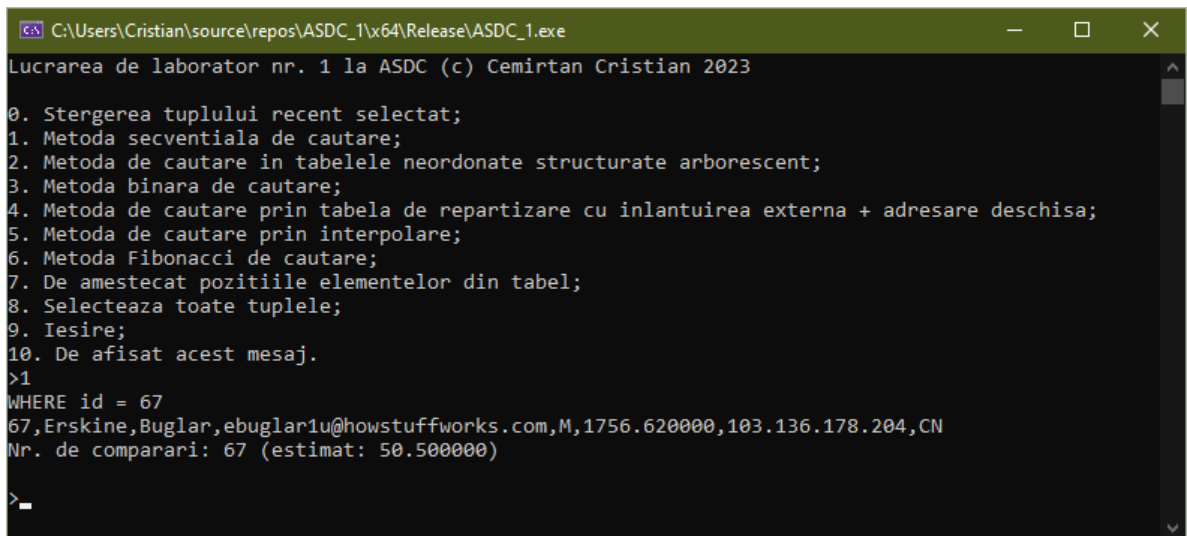
Baza de date menționată este stocată într-un singur fișier cu extensia .csv, care este apoi încărcat în memorie de către programul la momentul rulării.

Rularea metodelor de căutare

Metoda secvențială de căutare.

Se parcurge liniar de la primul element până la ultimul, până când se găsește elementul potrivit.

Este cea mai simplă metodă de căutare, dar cu cât mai multe înregistrări de date sunt într-un tabel, cu cât mai neeficientă metoda devine. Acest fapt este echivoc pentru căutarea unui element care se află mai departe de primul element.



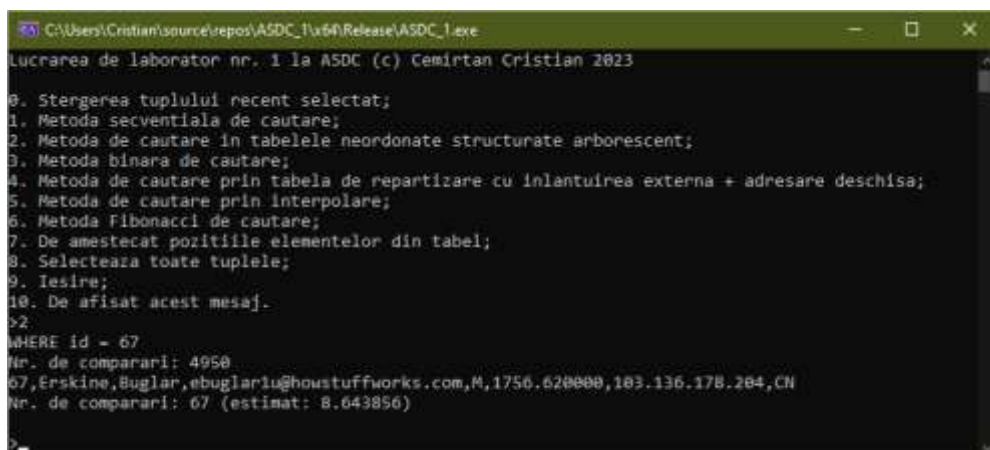
```
C:\Users\Cristian\source\repos\ASDC_1\Release\ASDC_1.exe
Lucrarea de laborator nr. 1 la ASDC (c) Cemirtan Cristian 2023

0. Stergerea tuplului recent selectat;
1. Metoda secventiala de cautare;
2. Metoda de cautare in tabelele neordonate structurate arborescent;
3. Metoda binara de cautare;
4. Metoda de cautare prin tabela de repartizare cu inlantuirea externa + adresare deschisa;
5. Metoda de cautare prin interpolare;
6. Metoda Fibonacci de cautare;
7. De amestecat pozitiile elementelor din tabel;
8. Selecteaza toate tuplele;
9. Iesire;
10. De afisat acest mesaj.
>1
WHERE id = 67
67,Erskine,Buglar,ebuglar1u@howstuffworks.com,M,1756.620000,103.136.178.204,CN
Nr. de comparari: 67 (estimat: 50.500000)
>
```

Figura 1

Metoda de căutare în tabelele neordonate structurate arborescent.

Pentru construirea arborelui binar de căutare, s-a necesitat un număr total de 4950 de comparații, iar lungimea de căutare rămâne identică cu metoda secvențială de căutare, pentru că inițial tabela era ordonată, și ca consecință am primit un arbore degenerat, sau un lanț.



```
C:\Users\Cristian\source\repos\ASDC_1\Release\ASDC_1.exe
Lucrarea de laborator nr. 1 la ASDC (c) Cemirtan Cristian 2023

0. Stergerea tuplului recent selectat;
1. Metoda secventiala de cautare;
2. Metoda de cautare in tabelele neordonate structurate arborescent;
3. Metoda binara de cautare;
4. Metoda de cautare prin tabela de repartizare cu inlantuirea externa + adresare deschisa;
5. Metoda de cautare prin interpolare;
6. Metoda Fibonacci de cautare;
7. De amestecat pozitiile elementelor din tabel;
8. Selecteaza toate tuplele;
9. Iesire;
10. De afisat acest mesaj.
>2
WHERE id = 67
Nr. de comparari: 4950
67,Erskine,Buglar,ebuglar1u@howstuffworks.com,M,1756.620000,103.136.178.204,CN
Nr. de comparari: 67 (estimat: 8.643856)
>
```

Figura 2

În caz contrar, dacă tabela este neordonată, atunci construirea arborelui binar și căutarea unui element în acest arbore devin, în mod concomitent, mai eficiente – practic s-au înjumătățit lungimile de căutare. Putem spune că, cu cât mai echilibrat este arborele binar de căutare, cu cât mai eficient devin ambele procesele menționate.

```

C:\Users\Cristian\source\repos\ASDC_1\Release\ASDC_1.exe
92,Libbey,Bruinemann,lbruinemann2j@drupal.org,F,1212.130000,82.97.102.87,CZ
93,Rochella,Albisser,ralbisser2k@vistaprint.com,F,2106.120000,48.139.98.13,CN
94,Etan,Corkell,ecorkell12l@spotify.com,M,1832.250000,192.115.139.63,BG
95,Gabriele,Custard,gcustard2m@dell.com,M,4270.730000,151.57.55.0,US
96,Riordan,Grigorey,rgrigorey2n@usnews.com,M,1274.580000,228.29.96.135,MG
97,Martina,Slarke,mslarke2o@live.com,F,4199.590000,59.205.34.125,ID
98,Orelee,Andresser,oandresser2p@reddit.com,F,3910.240000,10.213.93.25,ZM
99,Teddie,de'-Ancy Willis,tdeancywillis2q@cnbc.com,M,281.470000,117.203.181.68,CO
100,Alice,Hallibone,ahallibone2r@squarespace.com,F,2940.750000,79.112.173.176,ID

>7
activat

>2
WHERE id = 67
Nr. de comparari: 2500
67,Erskine,Buglar,ebuglar1u@howstuffworks.com,M,1756.620000,103.136.178.204,CN
Nr. de comparari: 35 (estimat: 8.643856)

>_

```

Figura 3

Metoda binară de căutare.

Prin metoda binară de căutare, înțelegem că vom testa egalitatea elementului din mijloc cu cel dorit, sau vom repeta recursiv acest principiu pentru spațiul de căutare împărțit în jumătate, în caz contrar.

```

C:\Users\Cristian\source\repos\ASDC_1\Release\ASDC_1.exe
2. Metoda de cautare in tabelele neordonate structurate arborescent;
3. Metoda binara de cautare;
4. Metoda de cautare prin tabela de repartizare cu inlantuirea externa + adresare deschisa;
5. Metoda de cautare prin interpolare;
6. Metoda Fibonacci de cautare;
7. De amestecat pozitiile elementelor din tabel;
8. Selecteaza toate tuplele;
9. Iesire;
10. De afisat acest mesaj.

>3
WHERE id = 50
50,Bernadene,Dowding,bdowding1d@washingtonpost.com,F,3696.190000,234.128.20.198,CN
Nr. de comparari: 1 (estimat: 8.643856)

>3
WHERE id = 2
2,Kellie,Colles,kcolles1@reuters.com,F,1209.730000,99.153.26.131,TN
Nr. de comparari: 7 (estimat: 8.643856)

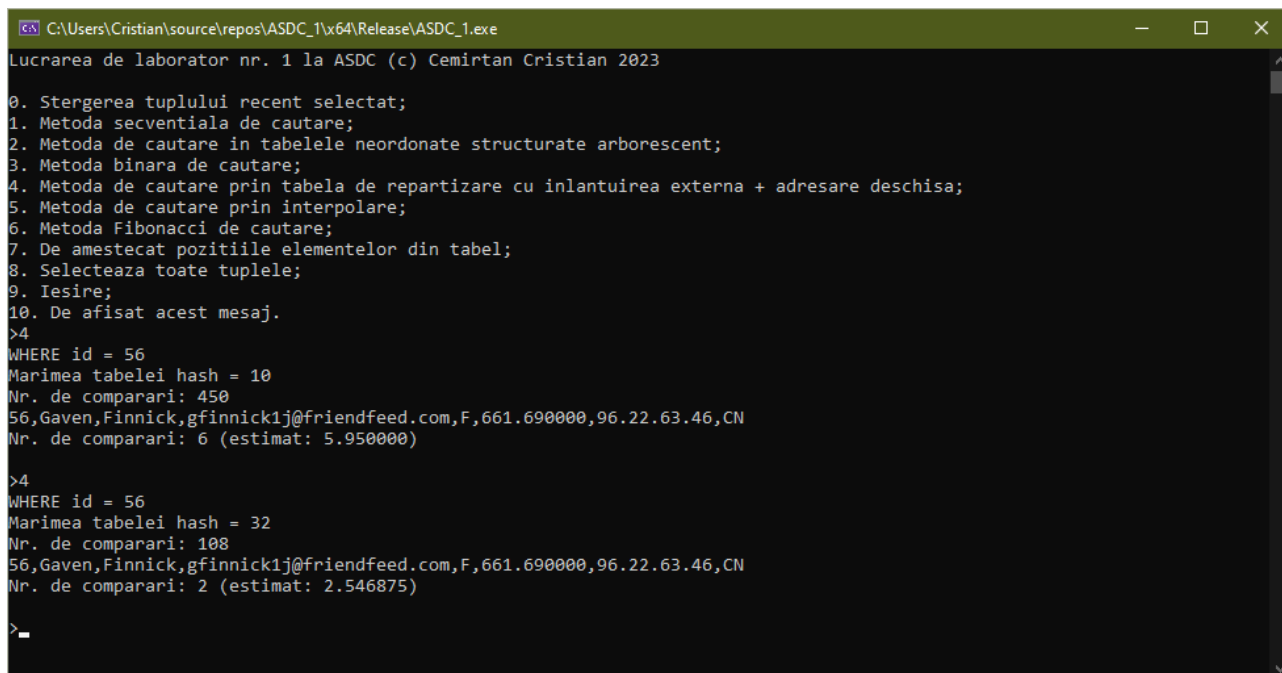
>_

```

Figura 4

Metoda de căutare prin tabela de repartizare.

Varianta tablei de repartizare este înlănțuirea externă împreună cu adresare deschisă. Implementarea va fi în felul următor: vom alocă dinamic un vector de reprezentare cu oricare lungime (introdus de la tastatură), și funcția de repartizare (hash) unei numere întregi va fi restul împărțirii sale la mărimea vectorului de reprezentare. Prin acest fapt, putem deduce că: șansa coliziunilor (nr. de elemente cu hash-ul identic) se reduce proporțional cu incrementarea lungimii vectorului de reprezentare.



```
C:\Users\Cristian\source\repos\ASDC_1\Release\ASDC_1.exe
Lucrarea de laborator nr. 1 la ASDC (c) Cemirtan Cristian 2023

0. Stergerea tuplului recent selectat;
1. Metoda secventiala de cautare;
2. Metoda de cautare in tabelele neordonate structurate arborescent;
3. Metoda binara de cautare;
4. Metoda de cautare prin tabela de repartizare cu inlantuirea externa + adresare deschisa;
5. Metoda de cautare prin interpolare;
6. Metoda Fibonacci de cautare;
7. De amestecat pozitiile elementelor din tabel;
8. Selecteaza toate tuplele;
9. Iesire;
10. De afisat acest mesaj.
>4
WHERE id = 56
Marimea tablei hash = 10
Nr. de comparari: 450
56,Gaven,Finnick,gfinnick1j@friendfeed.com,F,661.690000,96.22.63.46,CN
Nr. de comparari: 6 (estimat: 5.950000)

>4
WHERE id = 56
Marimea tablei hash = 32
Nr. de comparari: 108
56,Gaven,Finnick,gfinnick1j@friendfeed.com,F,661.690000,96.22.63.46,CN
Nr. de comparari: 2 (estimat: 2.546875)

>_
```

Figura 5

Metoda de căutare prin interpolare.

Numărul de comparații va fi 1 dacă pentru fiecare tuplu avem un ID care coincide cu poziția sa în tabelă. Numărul de comparații, în unele cazuri, poate fi mai mare decât 1 dacă condiția menționată nu se satisface.

Pe scurt, această metodă de căutare este eficientă dacă ID-urile tuplurilor au o distribuție uniformă între primul și ultimul element.

Conform formulei sale de calcul (vezi anexă), această metodă este compatibilă numai pentru calculele numerice, ceea ce înseamnă că ID-ul de căutat nu poate avea tipul de date diferit de un număr.

```
C:\Users\Cristian\source\repos\ASDC_1\64\Release\ASDC_1.exe
>5
WHERE id = 60
60,Fernanda,Lambard,flambard1n@cpanel.net,F,2784.510000,238.26.216.160,BR
Nr. de comparari: 1 (estimat: 0.301030)

>0
WHERE id = 60

>5
WHERE id = 61
61,Granthem,Tremain,gremain1o@salon.com,M,4970.310000,191.160.144.75,CZ
Nr. de comparari: 1 (estimat: 0.300081)

>0
WHERE id = 61

>5
WHERE id = 62
62,Phillida,Rudgerd,prudgerd1p@slashdot.org,F,2963.330000,50.23.2.184,CZ
Nr. de comparari: 1 (estimat: 0.299121)

>0
WHERE id = 62

>5
WHERE id = 59
59,Tore,Storrrar,tstorrrar1m@tuttocitta.it,M,4737.660000,197.194.227.32,PH
Nr. de comparari: 3 (estimat: 0.298148)

>_
```

Figura 6

Metoda Fibonacci de căutare.

Lungimea de căutare este eficientă dacă se cere de căutat un tuplu cu ID-ul care reprezintă un termen din șirul de numere Fibonacci.

```
WHERE id = 89
89,Avrit,Shelsher,ashelsher2g@alexa.com,F,595.020000,252.76.21.65,XK
Nr. de comparari: 1 (estimat: 8.643856)

>6
WHERE id = 55
55,Angelico,Lofts,alofts1i@sina.com.cn,M,3133.240000,92.192.22.157,ID
Nr. de comparari: 2 (estimat: 8.643856)

>6
WHERE id = 92
92,Libbey,Bruinemann,lbruinemann2j@drupal.org,F,1212.130000,82.97.102.87,CZ
Nr. de comparari: 5 (estimat: 8.643856)

>_
```

Figura 7

Lungimile medii de căutare teoretice pentru fiecare metodă de căutare

- Metoda secvențială de căutare: $(n + 1) / 2 = 50,5$;
 - Metoda de căutare în tabelele neordonate structurate arborescent: $\log_2(n) + 2 = 8,64$;
 - Metoda binară de căutare: $\log_2(n) + 2 = 8,64$;
 - Metoda de căutare prin tabela de repartizare cu înlănțuirea externă împreună cu adresarea deschisă: $1 + (n - 1) / (2 * m) = 1 + 99 / (2 * m)$;
 - Metoda de căutare prin interpolare: $\lg(\lg(n)) = 0,301$;
 - Metoda Fibonacci de căutare: $\log_2(n) + 2 = 8,64$;
- ✓ Unde: n – numărul inițial de tupluri în tabel; m – lungimea vectorului de reprezentare.

Concluzii

Conform experimentelor realizate, pot să concluzionez că:

- **Metoda de căutare în tabela de repartizare** este cea mai eficientă pentru tabelele neordonate. Cu cât mai mare este vectorul de repartizare, cu cât mai eficient devine căutarea unui element în tabel, dar crește și utilizarea memoriei operative.
- **Metoda de căutare prin interpolare** a fost demonstrată (în mod experimental) că are lungimile de căutare mai mici decât cele observate în metoda binară de căutare, deci este cea mai eficientă metodă de căutare în tabelele ordonate.
- **Metoda Fibonacci de căutare** are lungimile de căutare mai mari decât cele observate în metoda binară de căutare, dar nici într-un caz nu utilizează operațiile de împărțire, care este un avantaj masiv pentru sistemele de calcul unde o operație de împărțire este costisitoare de realizat.

Anexe

Se anexează fișierul .csv și codurile sursă elaborate în limbajul de programare C, în care sunt implementate metodele de căutare cerute în sarcină.