UNIVERSITATEA [NUMELE UNIVERSITĂȚII]
FACULTATEA DE MATEMATICĂ ŞI INFORMATICĂ

Specializarea Informatică

REFERAT

Disciplina: Algoritmi și structuri de date

Sortarea prin inserție și sortarea prin interclasare: comparație

Cadru didactic:
Lect. univ. dr. Bogdan Ilie

Student:
Vlad Georgescu
Anul I, grupa 106

Timișoara, 2026

CUPRINS

Introducere....3

1. Problema sortării şi măsurarea eficienţei....3

2. Cei doi algoritmi....3

2.1. Sortarea prin inserţie....3

2.2. Sortarea prin interclasare....4

3. Comparaţie pe şiruri generate....4

Concluzii....5

Bibliografie....5

Exemplu fictiv · licenta-disertatie-romania.com

Introducere

Sortarea, adică aşezarea elementelor unui şir într-o ordine dată, este una dintre cele mai studiate probleme din informatică. Ea apare peste tot: în listele ordonate alfabetic, în clasamente sau ca pas pregătitor pentru căutarea rapidă într-un şir.

Referatul compară doi algoritmi clasici de sortare, sortarea prin inserţie şi sortarea prin interclasare. Primul capitol defineşte problema şi modul în care se măsoară eficienţa unui algoritm, al doilea descrie cei doi algoritmi, iar al treilea îi compară pe baza numărului de comparaţii, numărate pe şiruri generate de program.

1. Problema sortării şi măsurarea eficienţei

Problema sortării primeşte la intrare un şir de n elemente şi cere o permutare a lor în ordine crescătoare (Cormen et al., 2022). Pentru algoritmii care sortează prin comparaţii, o măsură naturală a efortului este numărul de comparaţii între elemente, pentru că nu depinde de calculatorul pe care rulează programul.

Eficienţa se descrie prin felul în care creşte numărul de operaţii odată cu n. Notaţia O arată ordinul de creştere: un algoritm în O(n²) face, pentru un şir de zece ori mai lung, de aproximativ o sută de ori mai multe operaţii, în timp ce unul în O(n log n) creşte doar puţin mai repede decât liniar (Cormen et al., 2022; Knuth, 1998).

2. Cei doi algoritmi

2.1. Sortarea prin inserţie

Sortarea prin inserţie procedează ca un jucător care îşi aranjează cărţile în mână: ia elementele pe rând şi îl inserează pe fiecare la locul lui în partea deja sortată, deplasând spre dreapta elementele mai mari (Cormen et al., 2022).

```
pentru i ← 2, n execută
    cheie ← a[i]; j ← i − 1
    cât timp j ≥ 1 şi a[j] > cheie execută
        a[j + 1] ← a[j]; j ← j − 1
    a[j + 1] ← cheie
```

Pe un şir deja sortat, fiecare element se compară o singură dată, deci algoritmul face n − 1 comparaţii. Pe un şir sortat invers, fiecare element trece de toate cele dinaintea lui, deci

Exemplu fictiv · licenta-disertatie-romania.com

numărul de comparații este $n(n-1)/2$, de ordinul $O(n^2)$. Algoritmul sortează pe loc, fără memorie suplimentară, și este stabil: elementele egale își păstrează ordinea.

2.2. Sortarea prin interclasare

Sortarea prin interclasare aplică metoda divide et impera: împarte șirul în două jumătăți, le sortează separat, prin același procedeu, apoi le interclasează, comparând la fiecare pas primele elemente rămase din cele două jumătăți (Cormen et al., 2022; Sedgewick & Wayne, 2011).

```
sortare(a, st, dr):
    dacă st < dr atunci
        m ← (st + dr) div 2
        sortare(a, st, m); sortare(a, m + 1, dr)
        interclasare(a, st, m, dr)
```

Indiferent de ordinea inițială a elementelor, algoritmul face un număr de comparații de ordinul $O(n \log n)$. Prețul este memoria: interclasarea are nevoie de un șir auxiliar de aceeași lungime. Și această sortare este stabilă.

3. Comparație pe șiruri generate

Pentru a vedea diferența în practică, programul care a generat acest referat a rulat ambii algoritmi pe șiruri de numere generate pseudoaleator, de 10, 100 și 1.000 de elemente, și a numărat comparațiile. Pentru referință, tabelul arată și valorile $n^2/4$ (numărul mediu aproximativ de comparații la inserție) și $n \log_2 n$ (Tabelul 1).

Tabelul 1. Numărul de comparații pe șiruri pseudoaleatoare

n	Inserție: comparații	$n^2 / 4$	Interclasare: comparații	$n \log_2 n$
10	36	25	24	33
100	2.326	2.500	548	664
1.000	253.137	250.000	8.727	9.966

Sursa: comparații numărate de program pe șiruri generate pseudoaleator (exemplu); pe alte șiruri valorile diferă puțin.

La 10 elemente diferența este mică (36 față de 24 de comparații). La 1.000 de elemente, sortarea prin inserție face 253.137 de comparații, de aproximativ 29 de ori mai multe decât sortarea prin interclasare (8.727). Valorile urmează ordinele de creștere din teorie: la inserție se apropie de $n^2/4$, la interclasare rămân sub $n \log_2 n$.

Exemplu fictiv · licenta-disertatie-romania.com

Ordinea inițială contează mult la inserție și puțin la interclasare. Pe un șir de 1.000 de elemente deja sortat, sortarea prin inserție face doar 999 de comparații, iar pe același șir sortat invers 499.500. Sortarea prin interclasare face 4.932, respectiv 5.044 de comparații.

Concluzii

Cei doi algoritmi rezolvă aceeași problemă cu costuri diferite. Sortarea prin inserție este simplă, nu are nevoie de memorie suplimentară și este foarte rapidă pe șiruri scurte sau aproape sortate, dar numărul ei de comparații crește cu pătratul lungimii șirului. Sortarea prin interclasare păstrează un număr de comparații de ordinul n log n în orice situație, cu prețul unui șir auxiliar.

Alegerea depinde, așadar, de date: pentru șiruri mici sau aproape ordonate, inserția; pentru șiruri mari, interclasarea. Cormen et al. (2022) propun, ca exercițiu, chiar o variantă combinată, în care interclasarea împarte șirul, iar bucățile mici sunt sortate prin inserție.

Bibliografie

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). *Introduction to algorithms* (4th ed.). MIT Press.

Knuth, D. E. (1998). *The art of computer programming: Vol. 3. Sorting and searching* (2nd ed.). Addison-Wesley.

Sedgewick, R., & Wayne, K. (2011). *Algorithms* (4th ed.). Addison-Wesley.

Exemplu fictiv, pentru structură și formulare; nu îl preda ca atare.

Numele, datele din tabele și cazurile sunt inventate; sursele din bibliografie sunt reale. Model gratuit de pe licenta-disertatie-romania.com/resurse/model-referat-word/, unde găsești și șablonul gol.

Nu ai timp? Scriem referatul de la zero sau îl corectăm pe al tău, după cerințele profesorului. Livrare în maxim 2 zile, plată după livrare. Scrie-ne pe WhatsApp de pe licenta-disertatie-romania.com/servicii/referate-proiecte-facultate/