

Algoritmi de sortare

lucrare pentru atestarea
competențelor profesionale

elev: Alexuțan Cristian

clasa: XII D

profesor coordonator: Revnic Gheorghe

2024

Cuprins

Tema abordată	3
Detalii tehnice	4
Modul de utilizare	6
Bibliografie	8

Tema abordată

Am ales această temă deoarece sortarea unui șir este o problemă des întâlnită în algoritmică, ramură a informaticii studiată în cadrul liceului. Deseori, înainte de a aplica un algoritm asupra unui set de date, sortarea acestuia este necesară, un foarte bun exemplu fiind căutarea binară.

Astfel, pentru a îndeplini această cerință, există numeroase metode, de diferite complexități. Cele acoperite în cadrul site-ului sunt: metoda bulelor, sortarea prin selecție, sortarea prin inserare, sortarea prin interclasare și QuickSort, acestea fiind unele dintre cele mai des întâlnite modalități de sortare.

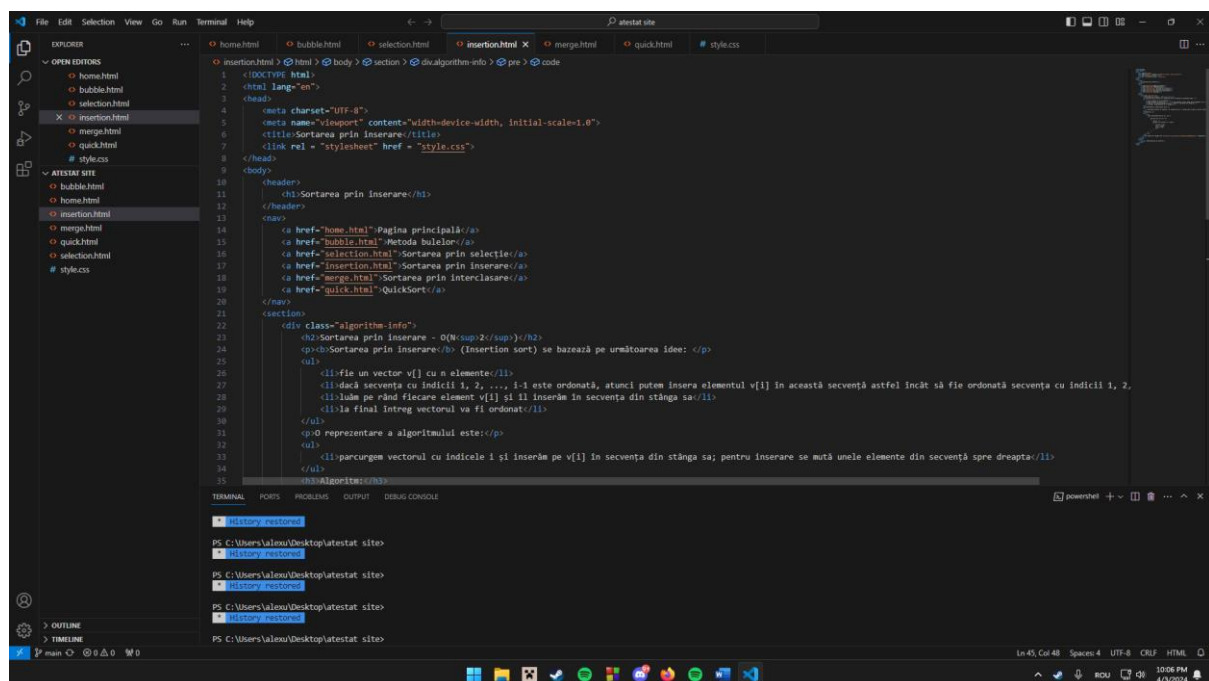
Detalii tehnice

Codul ce stă la baza site-ului a fost redactat în HTML și CSS, utilizând aplicația Visual Studio Code.

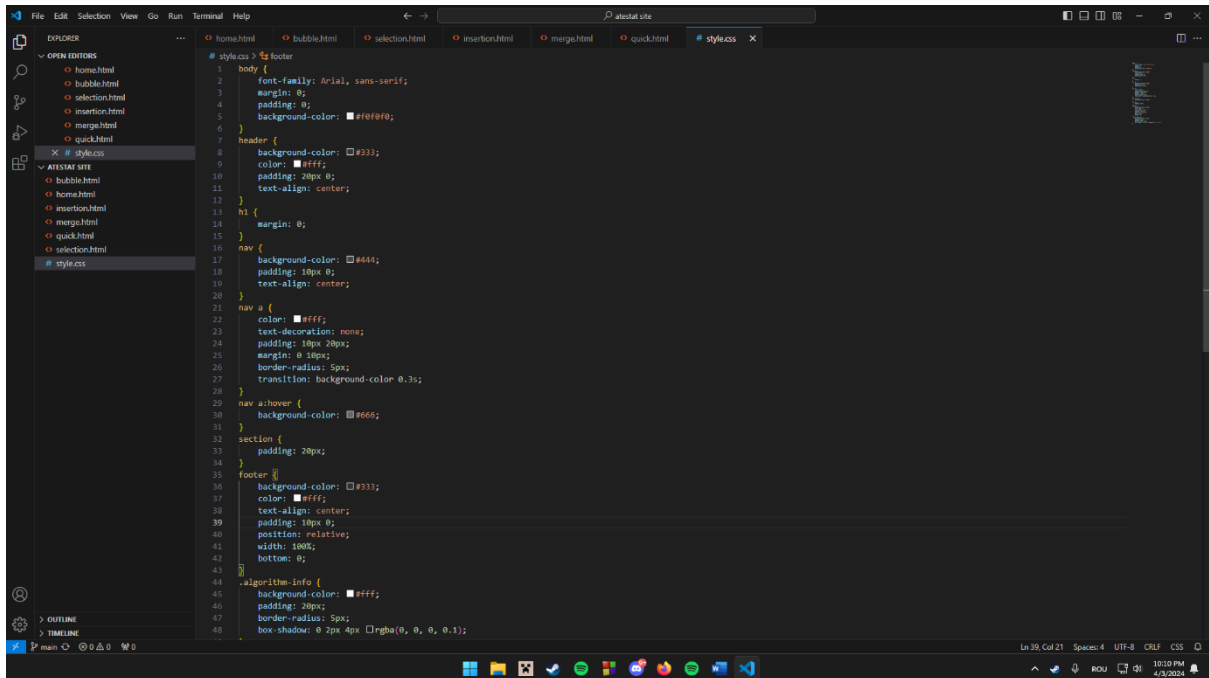
HyperText Markup Language (HTML) este un limbaj de marcare utilizat pentru crearea paginilor web ce pot fi afișate într-un browser (sau navigator). Scopul HTML este mai degrabă prezentarea informațiilor – paragrafe, fonturi, tabele ș.a.m.d. – decât descrierea semanticii documentului. În cadrul dezvoltării web de tip front-end, HTML este utilizat împreună cu CSS și JavaScript.

CSS sau Cascading Style Sheets este un standard pentru formatarea elementelor unui document HTML. Stilurile se pot atașa elementelor HTML prin intermediul unor fișiere externe sau în cadrul documentului, prin elementul `<style>` și/sau atributul `style`. CSS se poate utiliza și pentru formatarea elementelor XHTML, XML și SVG. CSS este una dintre tehnologiile de bază utilizate în procesul de dezvoltare web, împreună cu HTML și JavaScript.

Visual Studio Code, deseori numit VS Code, este un editor de cod creat de Microsoft pentru Linux, Windows și macOS.



În printscreen-ul de mai sus este regăsită interfața Visual Studio Code în timpul editării unei pagini din cadrul site-ului web. La începutul fiecărei pagini, în secțiunea <head> este definit style-ul pe care îl utilizează pagina, prin legătura cu fișierul extern style.css, din poza următoare:

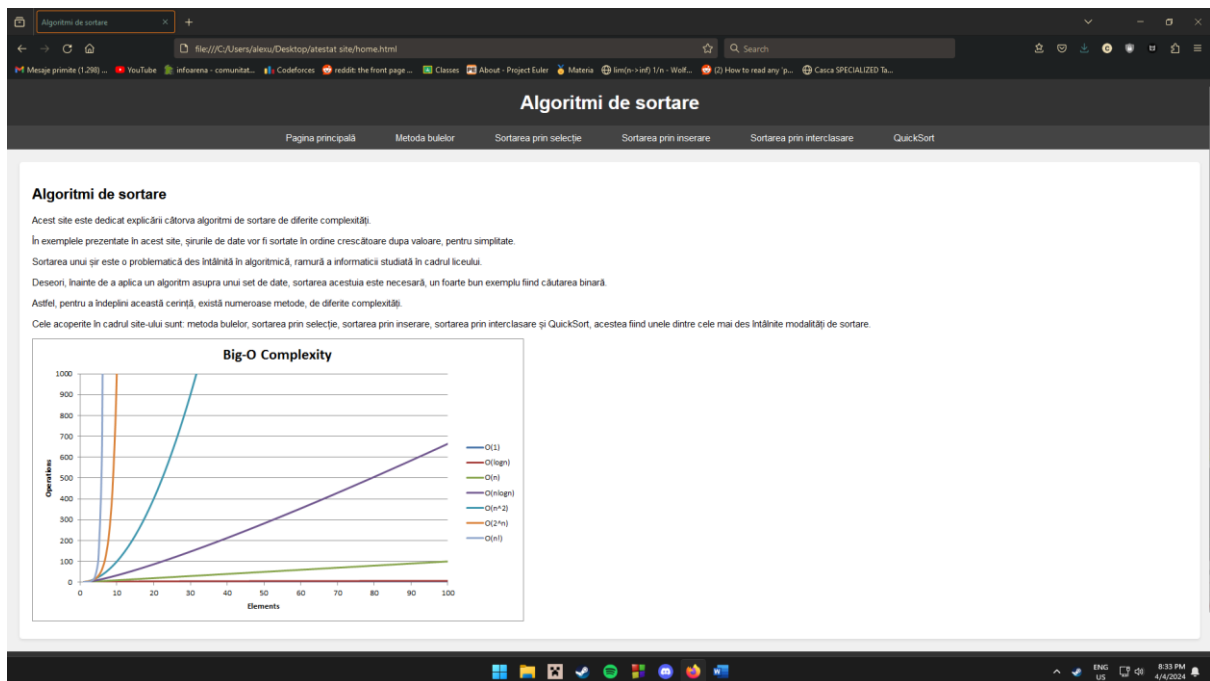


Aici sunt definite diferite elemente vizuale ale paginilor web.

Modul de utilizare

Interfața site-ului este una simplă. Fiecare pagină are în partea de sus o bară de navigație prin intermediul căreia putem trece de la o pagină la cealaltă. De asemenea, în partea de jos a fiecărei pagini este regăsit footer-ul, definit în style.css.

Pe fiecare pagină referitoare la un algoritm de sortare este prezent un video ce conține o reprezentare vizuală a algoritmului și a modului în care funcționează. Acestea au fost include în pagină prin intermediul funcției <embed> din HTML, de pe site-ul YouTube.



Sortarea prin interclasare

file:///C:/Users/alexu/Desktop/atestat/site/merge.html

Sortarea prin interclasare

Pagina principală Metoda bulelor Sortarea prin selecție Sortarea prin inserare Sortarea prin interclasare QuickSort

Sortarea prin interclasare - $O(N \log_2 N)$

Sortarea prin interclasare, sau **Mergesort** este o metodă eficientă de sortare a elementelor unui tablou, bazată pe următoarea idee:

Dacă prima jumătate a tabloului are elementele sortate și a doua jumătate are de asemenea elementele sortate, prin interclasare se va obține tabloul sortat.

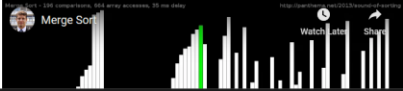
Sortarea prin interclasare este un exemplu tipic de algoritm divide et impera: se sortează o secvență delimitată de indici st și dr:

- dacă st == dr, problema este elementară, secvența fiind deja sortată
- dacă st < dr
 - se împarte problema în subprobleme, identificând mijlocul secvenței, m = (st + dr) / 2
 - se rezolvă subprobleme, sortând secvența delimitată de st și m, respectiv secvența delimitată de m+1 și dr - apeluri recursive
 - se combină soluțiile, interclasând cele două secvențe; în acest fel, secvența delimitată de st și dr este sortată

Algorithm:

```
int v[max+1];

void mergeSort(int left, int right)
{
    if(left == right) return;
    int mid = (left + right) / 2;
    mergeSort(left, mid);
    mergeSort(mid+1, right);
    int l = left, j = mid + 1;
    vector<int> temp;
    while(l <= mid and j <= right)
        if(v[l] < v[j]) temp.emplace_back(v[l++]);
        else temp.emplace_back(v[j++]);
    while(l <= mid) temp.emplace_back(v[l++]);
    while(j <= right) temp.emplace_back(v[j++]);
    for(int i = l; i <= right; i++) v[i] = temp[i-l];
}
```



10:17 PM 4/9/2024

Sortarea prin inserare

file:///C:/Users/alexu/Desktop/atestat/site/insertion.html

Sortarea prin inserare

Pagina principală Metoda bulelor Sortarea prin selecție Sortarea prin inserare Sortarea prin interclasare QuickSort

Sortarea prin inserare - $O(N^2)$

Sortarea prin inserare (Insertion sort) se bazează pe următoarea idee:


- fie un vector $v[i]$ cu n elemente
- dacă secvența cu indici 1, 2, ..., i-1 este ordonată, atunci putem insera elementul $v[i]$ în această secvență astfel încât să fie ordonată secvența cu indici 1, 2, ..., i-1, i
- luăm pe rând fiecare element $v[i]$ și îl inserăm în secvența din stânga sa
- la final întreg vectorul va fi ordonat

O reprezentare a algoritmului este:

- parcurgem vectorul cu indicii i și inserăm pe $v[i]$ în secvența din stânga sa, pentru inserare se mută unele elemente din secvență spre dreapta

Algorithm:

```
void sortareInserare(int v[], int n)
{
    for(int i=2; i<=n; i++)
    {
        int p = i;
        while(p > 0 and v[p] < v[p-1])
        {
            int aux = v[p];
            v[p] = v[p-1];
            v[p-1] = aux;
            p--;
        }
    }
}
```



ENG US 8:35 PM 4/9/2024

Bibliografie

1. <https://www.pbinfo.ro/>
2. https://ro.wikipedia.org/wiki/Cascading_Style_Sheets
3. https://ro.wikipedia.org/wiki/HyperText_Markup_Language
4. <http://elf.cs.pub.ro/sda-ab/wiki/laboratoare/laborator-02>
5. <https://www.youtube.com/@TimoBingmann>