SVEUČILIŠTE U ZAGREBU

**FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA**

Računalna Animacija

**Vizualizacija algoritama za sortiranje**

Josip Šestak 0036518569

Zagreb, siječanj 2023.

**Sadržaj**

[**Opis** 3](#_Toc124901564)

[Uvod 3](#_Toc124901565)

[O sortiranju 4](#_Toc124901566)

[Projekt 5](#_Toc124901567)

[**Upute za pokretanje** 8](#_Toc124901568)

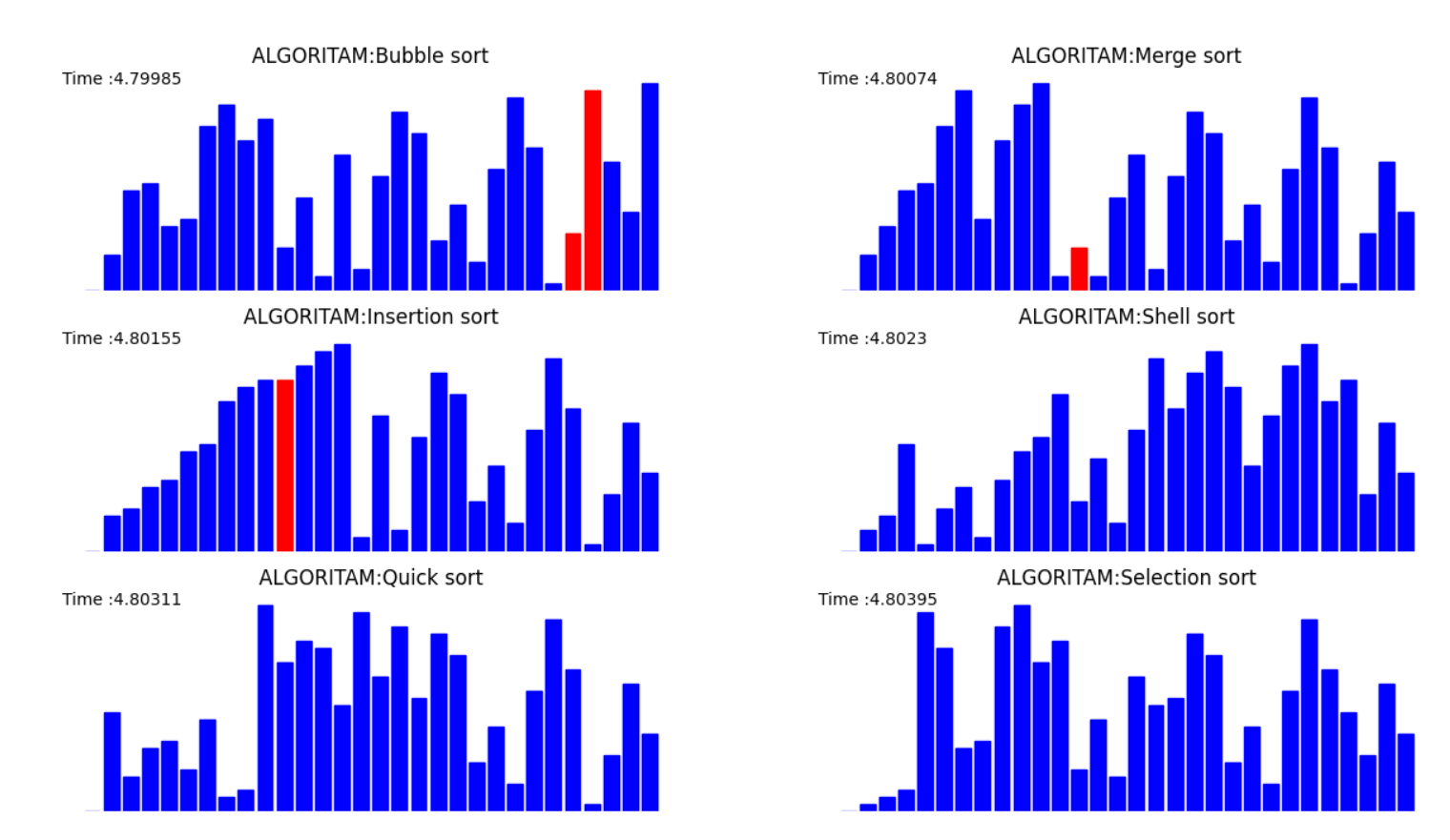
[**Sažetak** 9](#_Toc124901569)

[**Summary** 9](#_Toc124901570)

# **Opis**

## Uvod

Algoritmi sortiranja kao što i samo ime govori,koriste se za sortiranje liste podataka prema određenoj vrijednosti. Takvi algoritmi su veoma često korišteni u računarstvu. Ideja ovog rada je vizualizacija različitih algoritama sortiranja liste prirodnih brojeva gdje je visina svakog elementa liste proporcionalna njegovoj vrijednosti.



Slika 1

Algoritmi sortiranja koji su korišteni u radu su „bubble sort“, “merge sort“, “insertion sort“, “shell sort“, “quick sort“ i „selection sort“. Postoje mnogi drugi algoritmi, ali oni nisu korišteni.

## O sortiranju

Algoritmi sortiranja se opisuju prema nekoliko kriterija. Najvažniji kriterij u ovom radu koji je vidljiv pomoću animacija vizualizacije je vremenska složenost. Ostali kriteriji su memorijska složenost i stabilnost.

Vremenska složenost govori koliko je algoritam brz odnosno spor u odnosu na broj elemenata, dok nam memorijska složenost govori koliko memorije algoritam koristi u odnosu na broj elemanata.

Stabilnost algoritma govori hoće li poredak elemenata istih vrijednosti biti jednak nakon sortiranja.

Table

Description automatically generated

Slika 2

Slika 2 preuzeta je s [3].

Slika 2 prikazuje značajke svakog korištenog algoritma sortiranja.

## Projekt

Kod ovog rada je pisan u programskom jeziku Python te su korištene numpy i matplotlib biblioteke.

Vizualizacija se obavlja nad listom prirodnih brojeva. Zadana lista brojeva može biti učitana iz .txt dokumenta u koji je formatiran na način da svaki broj zauzima jedan redak ili može biti nasumično generirana. U drugom slučaju korisnik upisuje broj elemenata te se lista nasumično generira pomoću numpy biblioteke.

Kod može vizualizirati svaki od algoritama sortiranja nad listom zasebno ili može vizualizirati svih šest algoritama parelelno u jednom prozoru.

Visine stupova dijagrama su proporcionalne vrijednosti elemenata na tom mjestu te se stupovi zamjenjuju dok nisu sortirani po visini odnosno prema vrijednosti.

Crveno označeni stup u dijagramu označava element koji se trenutno mijenja.

Text

Description automatically generated

Slika 3 Chart, bar chart, histogram

Description automatically generated

Slika 4

Slika 3 prikazuje pokretanje programa iz naredbenog redka s nasumičnom listom od 30 elemenata i algoritmom bubble sort,a Slika 4 prikazuje sliku koja nastaje pokretanjem programa iz Slike 1.

Text

Description automatically generated

Slika 5

Chart

Description automatically generated

Slika 6

Slika 5 prikazuje pokretanje programa s nasumičnom listom od 30 elemenata i argumentom „compare“ koji prikazuje svih 6 algoritama paralelno, a Slika 6 prikazuje sliku koja takvim pokretanjem nastaje.

# **Upute za pokretanje**

Program se pokreće uz 2 argumenta.

Prvi argument je prirodni broj koji označava broj elemenata nasumično generirane liste ili .txt dokument koji sadrži elemente liste koji su postavljeni jedan za drugim u vlastitom retku.

Za pokretanje svih algoritama sortiranja paralelno , drugi argument je „compare“.

Za pokretanje pojedinog algoritma, drugi argument je naziv sorta malim slovima odvojen s „\_“. Npr. „bubble\_sort“ , „quick\_sort“ itd...

# **Sažetak**

Ovaj rad bavi se vizualizacijom algoritama sortiranja pomoću biblioteke matplotlib. Moguće je prikazati šest zadanih algoritama sortiranja odjednom ili svaki zasebno. Algoritmi sortiraju nasumično izmješanu listu proizvoljne duljine ili određenu listu zadanu .txt dokumentom.

# **Summary**

This work visualizes sorting algorithms using matplotlib library. The work is able to visualize six sorting algoritithms at once or each sorting algorithm separately.

Algorithms can sort randomly generated list of defined size or sort the list obtained by the .txt file.

# **Literatura**

1. Predmet Algoritmi i Strukture Podataka na Fakultetu Elektrotehnike i Računarstva. <https://www.fer.unizg.hr/predmet/asp_b>
2. Predmet Računalna Animacija na Fakultetu Elektrotehnike i Računarstva. <https://www.fer.unizg.hr/predmet/racani_b>
3. Poveznica: <https://www.geeksforgeeks.org/sorting-algorithms/>