

UNIVERSITETI I PRISHTINËS  
“HASAN PRISHTINA”  
FAKULTETI I INXHINIERISË ELEKTRIKE DHE KOMPJUTERIKE  
DEGA: INXHINIERI KOMPJUTERIKE



**Lënda: Dizajni dhe Analiza e Algortimeve**

**Tema: Radix Sort**

Prof. Dr. Avni Rexhepi

Asistent: Dardan Shabani

Emri dhe mbiemri	Nr. ID
Egzon Gashi	180715100044
Albin Semetishti	180715100015
Betim Tasholli	180715100014

## Përmbajtja

Historiku .....	3
Çka është Radix Sort? .....	4
Kompleksiteti i Radix Sort .....	4
Avantazhet dhe disavantazhet .....	4
Krahasimi i Radix Sort me algoritmet tjera të sortimit .....	5
Shembull .....	5
Graphical User Interface .....	6
Referencat .....	7

## Historiku

Historia e Radix Sort fillon në 1887 me punën e Herman Hollerith në makinat tabulative. Algoritmet Radix-Sort erdhën në përdorim si një mënyrë për të sortuar “punched cards” karta te vrimëzuara prej vitit 1923, Algoritmi I parë kompjuterik me memorje efikente u zhvillua në 1954 në MIT nga Harold H.Seward. Radix sortet e kompjuterizuara më parë ishin menduar si jopraktike për shkak të nevojës për alokimin e hapësirave të panjohura për variabla. Risia e Seward ishte të përdorej një skan linear për të përcaktuar hapësirën e kërkuar dhe offset-et më përpara, duke lejuar një alokim static të vetëm të memorjes. Skanimi linear është i lidhur ngusht me algoritmin tjetër të Seward – sortimi i numërimit. Në kohën e sotme Radix Sort përdoret më së shpeshti në mbledhjen e integjerëve dhe stringjeve binare. Është treguar të jetë më i shpejtë se algoritme të tjera më të përgjithshme të sortimit nganjëherë 50% deri në 3 herë më shpejtë.



Fig. 1. Një IBM card sorter duke përformuar radix sort-in.

## Çka është Radix Sort?

Në shkencat kompjuterike, radix sort është një algoritm jo-krahasues i sortimit. Eviton krahasimin duke krijuar dhe shpërndarë elemente në buckets sipas radiksit të tyre. Për elementët me më shumë se një shifër të konsiderueshme, ky proces përsëritet për secilën shifër, duke ruajtur renditjen e hapit paraprak, derisa të merren parasysh të gjitha shifrat.

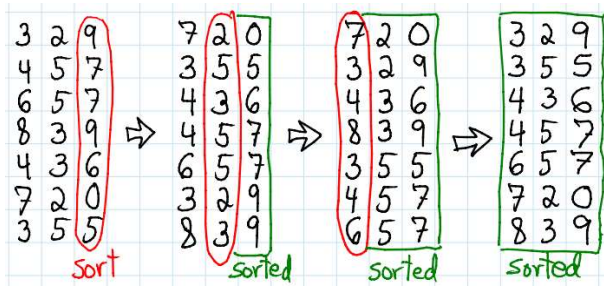


Fig. 2. Shembull i Radix Sort në fletë.

## Kompleksiteti i Radix Sort

Kompleksiteti më i keq:  $n * k / d$   
Kompleksiteti i hapësirës:  $n + 2^d$   
Kompleksiteti i kohës:  $d * (n + b)$

n - Numri i elementeve.

k - Vlera më e madhe e mundshme.

d - Numri i shifrave të dhëna në list.

b - Baza e numrit (P.Sh. 10 - Decimal).

## Avantazhet dhe disavantazhet

Avantazhet:

1. I shpejtë kur qelësat janë të shkurtë që d.m.th. kur rangi i elementeve të vektorit është më i vogël.
2. Përdoret në algoritmet e konstruktimit si algoritmi i Manber dhe algoritmi DC3.

Disavantazhet:

1. Pasi që Radix Sort varet nga shifrat është shumë më pak fleksibil sesa sortimet tjera. Pra, për cdo tip të ndryshëm të të dhënave ai duhet të rishkruhet.
2. Konstanta për Radix Sort është më e madhe krahasuar me algoritmet tjera të sortimit.
3. Merr më shumë hapësirë krahasuar me Quick-Sort që është sortim statik.

## Krahasimi i Radix Sort me algoritmet tjera të sortimit

Radix Sort edhe pse për numër të vogël të anëtarëve në listë është më i ngadaltë mirëpo me rritjen e numrit të anëtarëve shpejtësia e Radix Sort në krahasim me Quick-Sort rritet ndjeshëm.

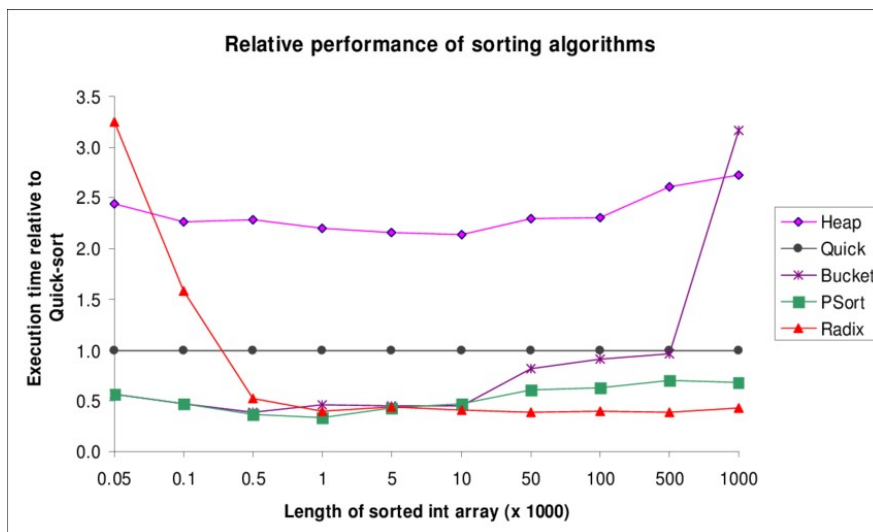


Fig. 3. Krahasimi i Quick-Sort me algoritmet e tjera përfshirë Radix Sort.

## Shembull

Lista e dhënë e pasortuar:

[180, 51, 76, 41, 3, 235, 88, 138]

Duke filluar nga shifra më e djathtë (e fundit), rendisim numrat bazuar në atë shifër:

[{180}, {51, 41}, {3}, {235}, {76}, {88, 138}]

Vazhdojmë sipas shifrës tjetër në të majtë:

[{03}, {235, 138}, {41}, {51}, {76}, {180, 88}]

Dhe së fundmi nga shifra më e majtë:

[{003, 041, 051, 076, 088}, {138, 180}, {235}]

Lista që sortuam është:

[3, 41, 51, 76, 88, 138, 180, 235].

## Graphical User Interface

Për programimin e këtij programi kemi përdorur gjuhën C#. Në text label duhet dhënë numrat të cilët dëshirojmë t'i sortojmë të ndarë me presje, pastaj klikojmë butonin “Sort them using radix sort.” dhe rezultati shfaqet poshtë butonit së bashku me kohën e ekzekutimit. Siç shihet në figurën më poshtë kemi marrur listen nga shembulli i mësipërm.

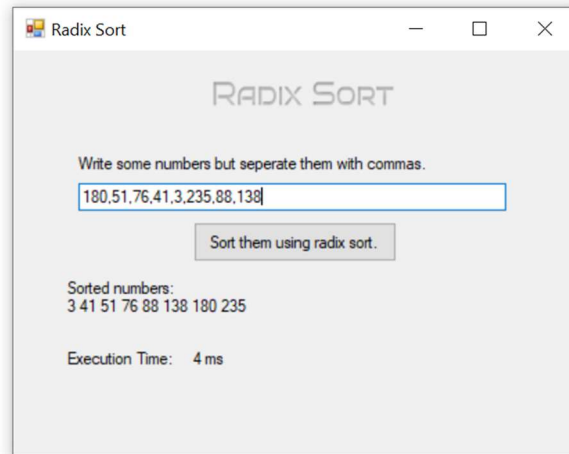


Fig. 4. Testimi i Radix Sort algoritmit.

## Referencat

Radix Sort. (n.d.). Retrieved from Wikipedia: [https://en.wikipedia.org/wiki/Radix\\_sort](https://en.wikipedia.org/wiki/Radix_sort)

Radix Sort. (n.d.). Retrieved from YouTube: [https://www.youtube.com/watch?v=XiuSW\\_mEn7g](https://www.youtube.com/watch?v=XiuSW_mEn7g)