

Kodutöö 2

Liisbeth Tatter

EDTR207877

Ülesanne 1: Bubble Sort simulatsioon

Antud on järgnevate arvude loend: [64, 34, 25, 12, 22, 11, 90].

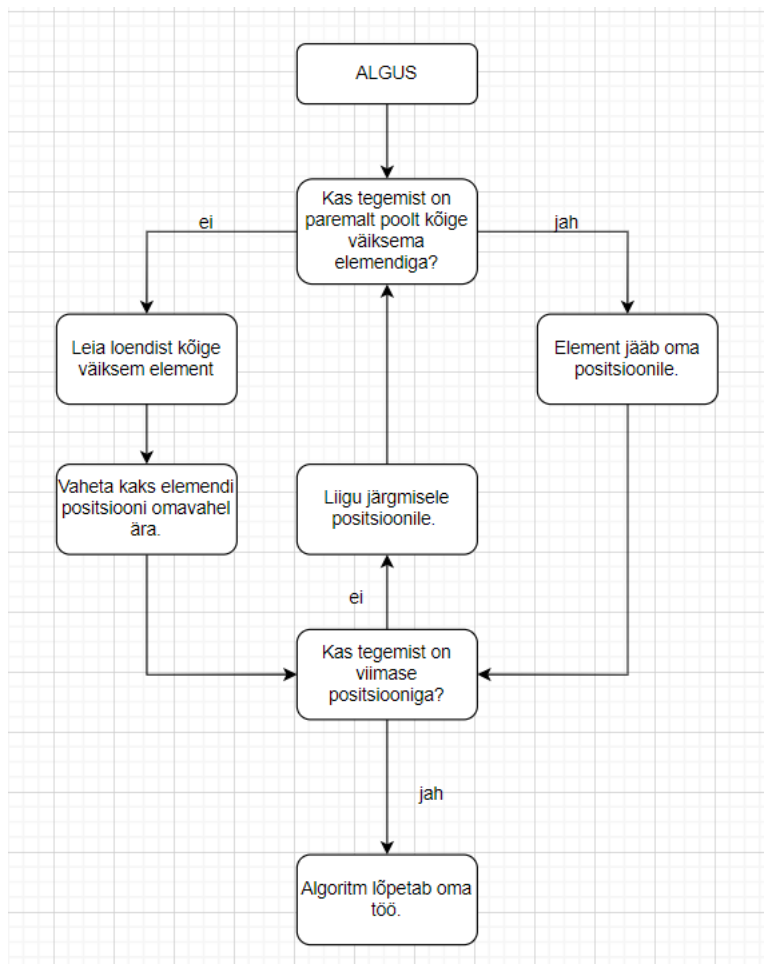
Simuleeri samm-sammult Bubble Sort algoritmi. Iga läbimise järel kirjuta üles tulemuseks olev loend.

64 > 34 seega asukohad vahetatakse	[64, 34, 25, 12, 22, 11, 90]
64 > 25 seega asukohad vahetatakse	[34, 64, 25, 12, 22, 11, 90]
64 > 12 seega asukohad vahetatakse	[34, 25, 64, 12, 22, 11, 90]
64 > 22 seega asukohad vahetatakse	[34, 25, 12, 64, 22, 11, 90]
64 > 11 seega asukohad vahetatakse	[34, 25, 12, 22, 64, 11, 90]
64 < 90 seega asukohti ei vahetata	[34, 25, 12, 22, 11, 64, 90]
34 > 25 seega asukohad vahetatakse	[34, 25, 12, 22, 11, 64, 90]
34 > 12 seega asukohad vahetatakse	[25, 34, 12, 22, 11, 64, 90]
34 > 22 seega asukohad vahetatakse	[25, 12, 34, 22, 11, 64, 90]
34 > 11 seega asukohad vahetatakse	[25, 12, 22, 34, 11, 64, 90]
34 < 64 seega asukohti ei vahetata	[25, 12, 22, 11, 34, 64, 90]
64 < 90 seega asukohti ei vahetata	[25, 12, 22, 11, 34, 64, 90]
25 > 12 seega asukohad vahetatakse	[25, 12, 22, 11, 34, 64, 90]
25 > 22 seega asukohad vahetatakse	[12, 25, 22, 11, 34, 64, 90]
25 > 11 seega asukohad vahetatakse	[12, 22, 25, 11, 34, 64, 90]
25 < 34 seega asukohti ei vahetata	[12, 22, 11, 25, 34, 64, 90]
34 < 64 seega asukohti ei vahetata	[12, 22, 11, 25, 34, 64, 90]
64 < 90 seega asukohti ei vahetata	[12, 22, 11, 25, 34, 64, 90]
12 < 22 seega asukohti ei vahetata	[12, 22, 11, 25, 34, 64, 90]
22 > 11 seega asukohad vahetatakse	[12, 22, 11, 25, 34, 64, 90]
22 < 25 seega asukohti ei vahetata	[12, 11, 22, 25, 34, 64, 90]
25 < 34 seega asukohti ei vahetata	[12, 11, 22, 25, 34, 64, 90]
34 < 64 seega asukohti ei vahetata	[12, 11, 22, 25, 34, 64, 90]
64 < 90 seega asukohti ei vahetata	[12, 11, 22, 25, 34, 64, 90]

11 < 12 seega asukohti ei vahetata	[11, 12, 22, 25, 34, 64, 90]
12 < 22 seega asukohti ei vahetata	[11, 12, 22, 25, 34, 64, 90]
22 < 25 seega asukohti ei vahetata	[11, 12, 22, 25, 34, 64, 90]
25 < 34 seega asukohti ei vahetata	[11, 12, 22, 25, 34, 64, 90]
34 < 64 seega asukohti ei vahetata	[11, 12, 22, 25, 34, 64, 90]
64 < 90 seega asukohti ei vahetata	[11, 12, 22, 25, 34, 64, 90]

Ülesanne 2: Selection Sort visualiseerimine (1p)

Joonista plokkskeem, mis esindab Selection Sort algoritmi. Kasutades seda vooluskeemi, demonstreeri sortimisprotsessi loendile: [29, 15, 56, 77, 18]. Tuvasta iga iteratsiooni ajal väikseim element.



[29, 15, 56, 77, 18]
[15, 29, 56, 77, 18]
[15, 18, 56, 77, 29]
[15, 18, 29, 77, 56]
[15, 18, 29, 56, 77]

Ülesanne 3: Insertion Sort praktikas (1p)

Antud on osaliselt sorteeritud loend: [12, 11, 13, 5, 6, 7]. Rakenda Insertion Sort algoritmi. Selgita, miks Insertion Sort võib olla tõhusam sellele loendile võrreldes täielikult sortimata loendiga.

[12, 11, 13, 5, 6, 7]
[12, 11, 13, 5, 6, 7]
[11, 12, 13, 5, 6, 7]
[11, 12, 13, 5, 6, 7]
[11, 12, 13, 5, 6, 7]
[5, 11, 12, 13, 6, 7]
[5, 6, 11, 12, 13, 7]
[5, 6, 7, 11, 12, 13]

Ülesanne 4: Ajakompleksuste analüüs (1p)

a) Millistel eelmainitud sortimisalgoritmidel on halvimal juhul ajakompleksus $O(n^2)$?
Bubble sort, selection sort, insertion sort ehk kõigil kolmel.

b) Milline sortimisalgoritm oleks kõige sobivam sortimaks loendit täisarvudega, mis jäävad vahemikku 1 kuni 100 ja miks?

Insertion sort

Ülesanne 5: Stabiilsus ja adaptiivsus sortimisel (1p)

a) Defineeri, mida tähendab, et sortimisalgoritm on "stabiilne." Anna näide stabiilsest sortimisalgoritmist antud loendist.

Stabiilsus sortimisalgoritmis tähendab võrdsete elementide suhtelise järjekorra säilitamist sorteeritud väljundis. Bubble sort on stabiilne, selection sort ei ole

b) Selgita "adaptiivsuse" kontseptsiooni sortimisel. Millistest eelmainitud sortimisalgoritmidest peetakse adaptiivseks?

Sortimisalgoritm on kohanduv, kui see kasutab ära juba sorteeritud elemente järjendis, mida tuleb sorteerida. Insertion sort on adaptiivne

Boonus (3p):

Kuidas näeks loend [8, 3, 5, 4, 7, 6, 2] välja pärast esimest läbimist antud algoritmide puhul

- a) Bubble Sort – [3, 8, 5, 4, 7, 6, 2]
- b) Selection Sort – [2, 3, 5, 4, 7, 6, 8]
- c) Insertion Sort – [3, 8, 5, 4, 7, 6, 2]