

Algoritmi za sortiranje

navadni algoritmi:

- navadno vstavljanje

princip:

```
for i:=2 to n do
  begin
    x:=a[i];
    vstavi x na pravo mesto med a[1] in a[I];
  end;
```

- izboljšava:

- binarno vstavljanje – hitrejše iskanje mesta kamor se vstavi element

- navadno izbiranje

princip:

```
for i:=1 to n-1 do
  begin
    poišči najmanjši elemnt med a[i]...a[n];
    zamenjaj najmanjši element in a[i];
  end;
```

- sortiranje s premenami – bubble sort

princip:

```
for i:=2 to n do
  begin
    od desne meje do I davzdol primerjaj 2 sosednja elementa in
    po potrebi zamenjaj;
  end;
```

- izboljšave:

- postopek sortiranja prekinemo, če v zadnjem prehodu ni bilo nobene zamenjave
- zapomnimo si mesto zamenjave v prejšnjem prehodu. V naslednjem prehodu s pomikamo od desne proti levi samo do mesta zadnje zamenjave
- menjamo smeri prehodov – shaker sort

izpopolnjeni algoritmi:

- shellsort

je izboljšava navadnega vstavljanja:

- uporabimo postopek navadnega vstavljanja v več etapah, vsakrat z različnim korakom
- zadnji korak mora biti 1
- pomembna je izbira korakov – doseči je treba čimboljše prepletanje

princip:

begin

 določi vrednost korakov – tabela h

 for m:=1 to t do

 begin

 izberi korak in inicializiraj položaj stražarja za to etapo

 for i:=k+1 to n do

 begin

 x:=a[i]; izberemo opazovani element

 nastavimo stražarja;

 vstavi x na pravo mesto med a[1] in a[i] upoštevajoč korak

 end;

 end;

 end;

- heapsort

princip:

1. vhodno zapredjeuredimo v kopico
2. zamenjamo 1. in zadnji element kopice (zadnji element pride na pravo mesto, velikost kopice se zmanjša za 1)
3. popravimo kopico
4. ponavljaj korak 2 in 3 dokler kopica ne skopni

- quicksort

osnova za quicksort je porazdelitveni algoritem, ki razdeli tabelo na dva dela.

princip:

1. izberemo poljubni element tabele (npr. srednji): x
2. pregledujemo tabelo z leve dokler ne naletimo na element, ki je večji ali enak od elementa x
3. pregledujemo z desne proti levi dokler ne naletimo na element, ki je manjši ali enak od elementa x
4. zamenjamo elementa, pri katerih sta se ustavili pregledovanji
5. ponavljamo korake 2, 3 in 4 dokler se pregledovanja ne srečata sredi tabele.