

1. Imel si celotno kodo heapsort. Spremeni program tako, da bo na koncu izpisal:
-kolikokrat so se elementi pogreznili po levi
-kolikokrat so se elementi pogreznili po desni
-povprečno pogrezanje

2. Simuliraj polifazno sortiranje nad seznamom. Elementi so urejeni po padajočem vrstnem redu, število trakov je 5 (N=5), po načinu, kot ste ga naredili na vajah.:

1 2 3 4 5 6 7 8 9 9 1 2 3 4 5 6 7 8 9 9 1 2

3. Napiši funkcijo Findbigger, ki bo v seznamu-urejenem po naraščajočem vrstnem redu, poiskala element ki je enak ali večji od el

seznam je realiziran s kazalci

```
ElementType=....;  
Plist=^list;  
list=record  
    element:ElementType;  
    next:^Plist;  
end;
```

position: Plist;

Function Findbigger(L:list; el:ElementType):position

Realizirati je bilo treba še eno proceduro, ki s pomočjo Findbigger vsavi element na pravo mesto tako, da bojo podatki ostali urejeni. Seznam je realiziran brez slepega elementa.

4. Nariši binarno iskalno drevo

10 11 8 7 6 .. (okrog 10 elementov)

-izpiši to drevo v obratnem redu

-izbriši iz njega 3 elemente (je bilo podano)

-po kakšnem vrstnem redu bi moral biti vnos podatkov, da bi bilo drevo izravnano