

## Programiranje 2 — drugi izpitni rok

### 26. junij 2023

Oddajte datoteke `naloga1.c`, `naloga2.c` in `naloga3.c`. Točkovanje: 30, 30, 40.

- ① Napišite program (`naloga1.c`), ki podano vhodno dvojiško datoteko prepíše v podano izhodno dvojiško datoteko, le da vsako zaporedje najmanj dveh enakih bajtov zamenja z enim samim takim bajtom. Ime vhodne datoteke je podano kot prvi, ime izhodne datoteke pa kot drugi argument ukazne vrstice. Vhodna datoteka vsebuje kvečjemu  $10^5$  bajtov.

Sledi primer vhodne in pripadajoče izhodne datoteke (v šestnajstiškem zapisu):

`test01.bin:`

AB AB AB D5 39 39 39 D5 D5 AB
-------------------------------

`test01.bout:`

AB D5 39 D5 AB
----------------

- ② Napišite program (`naloga2.c`), ki s standardnega vhoda prebere niz dolžine  $n \in [1, 15]$  ter pozitivni celi števili  $a \in [1, n]$  in  $b \in [a, n]$ , na standardni izhod pa izpiše vsa mogoča razbitja vhodnega niza na najmanj  $a$  in največ  $b$  nepraznih podnizov. Razbitja lahko izpišete v poljubnem vrstnem redu. Vhodni niz je sestavljen zgolj iz velikih in malih črk angleške abecede.

Sledi primer:

`test01.in:`

zanka 2 3
-----------

`test01.out:`

z a nka z an ka z ank a z anka za n ka za nk a za nka zan k a zan ka zank a
--

Za delne točke:

- V 50% testnih primerov velja  $a = 1$  in  $b = n$ .

- ③ V datoteki `naloga3.h` je definiran tip `Vozlisce`, ki omogoča povezovanje vozlišč v pravokotno mrežo:

```
typedef struct _Vozlisce Vozlisce;
struct _Vozlisce {
    int vsebina;           // »vsebina« vozlišča
    Vozlisce* desno;       // kazalec na desnega sosedo (NULL v zadnjem stolpcu)
    Vozlisce* dol;         // kazalec na spodnjega sosedo (NULL v zadnji vrstici)
};
```

Naj bo  $h$  višina,  $w$  pa širina mreže. V vseh testnih primerih velja  $h, w \in [1, 100]$ .

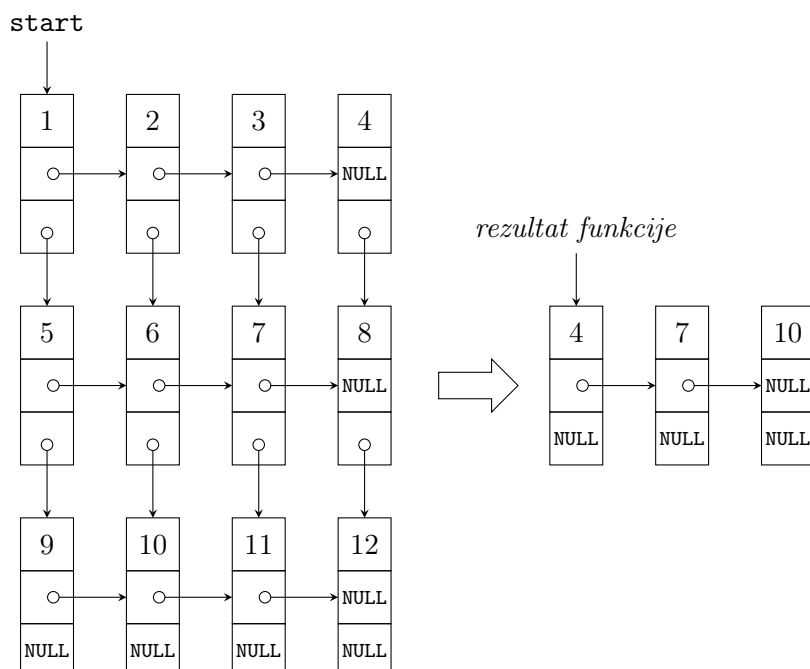
V datoteki `naloga3.c` dopolnite funkcijo

```
Vozlisce* diagonalna(Vozlisce* start, int* vsota),
```

ki na podlagi diagonale mreže, ki se začne v zgornjem desnem vozlišču, konča pa bodisi na spodnjem bodisi na levem robu (odvisno od oblike mreže), izdelava *nov* povezani seznam, v katerem imajo vozlišča isto vsebino kot vozlišča na opisani diagonali. Vozlišča v seznamu naj bodo povezana prek kazalcev `desno`; kazalci `dol` naj bodo vsi enaki NULL. Funkcija naj poleg tega v spremenljivko, na katero kaže kazalec `vsota`, vpiše vsoto vsebin vozlišč diagonale, vrne pa naj kazalec na prvo vozlišče izdelanega seznama.

Funkcija naj vhodno mrežo pusti nedotaknjeno.

V sledečem primeru (`test01.c`) po opisanem receptu izdelamo povezani seznam za mrežo višine 3 in širine 4:



Za delne točke:

- V 20% testnih primerov velja  $h = 1$ .
- V nadaljnjih 30% testnih primerov preverjamo samo vrednost spremenljivke, na katero kaže kazalec `vsota`, kazalec, ki ga funkcija vrne, pa ignoriramo.