

Programiranje 2 — tretji izpitni rok

21. avgust 2023

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

- ① Napišite program (`naloga1.c`), ki na podlagi dveh vhodnih dvojiških datotek (denimo, da prva vsebuje a , druga pa b bajtov) izdelava izhodno dvojiško datoteko z $\max(a, b)$ bajti, pri čemer naj bo i -ti bajt izhodne datoteke (za vse $i \in \{1, \dots, \min(a, b)\}$) enak večjemu izmed i -tega bajta prve vhodne datoteke in i -tega bajta druge vhodne datoteke, preostali bajti izhodne datoteke pa naj bodo enaki preostalim bajtom daljše izmed obeh vhodnih datotek.

Imeni vhodnih datotek sta podani kot prvi in drugi argument ukazne vrstice, ime izhodne datoteke pa kot tretji argument.

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

`test01a.bin:`

CB	0A	62	9D	F3
----	----	----	----	----

`test01b.bin:`

4B	90	41	9D	C8	81	53
----	----	----	----	----	----	----

`test01.bout:`

CB	90	62	9D	F3	81	53
----	----	----	----	----	----	----

V vseh testnih primerih velja $a, b \in [0, 10^5]$. V 30% testnih primerov velja $a = b > 0$, v nadaljnjih 30% primerov pa $a > b > 0$. V 50% testnih primerov je vsak bajt enak bodisi 0 bodisi 1.

- ② Slika velikosti $n \times m$ pik je v pomnilniku predstavljena s kazalcem na začetek tabele n kazalcev, od katerih vsak kaže na začetek tabele m celih števil z intervala $[0, 2023]$. Ta števila predstavljajo sivine posameznih pik.

V datoteki `naloga2.c` dopolnite funkcijo

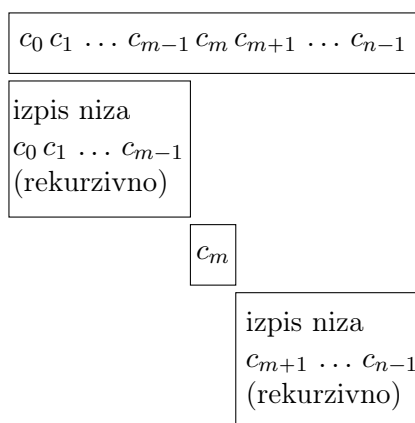
```
int** zmehcaj(int** slika, int n, int m, int d),
```

ki vrne novo sliko, ki je v pomnilniku predstavljena na enak način, le da je velikosti $(n - 2d) \times (m - 2d)$ in da je sivina vsake pike (i, j) nove slike povprečje svin vseh svin pik v d -okolici stare pike $(i + d, j + d)$, pri čemer je d -okolica pike $(i + d, j + d)$ kvadrat s stranico dolžine $2d + 1$ in središčem $(i + d, j + d)$. Pri računanju povprečja se zaokrožuje navzdol.

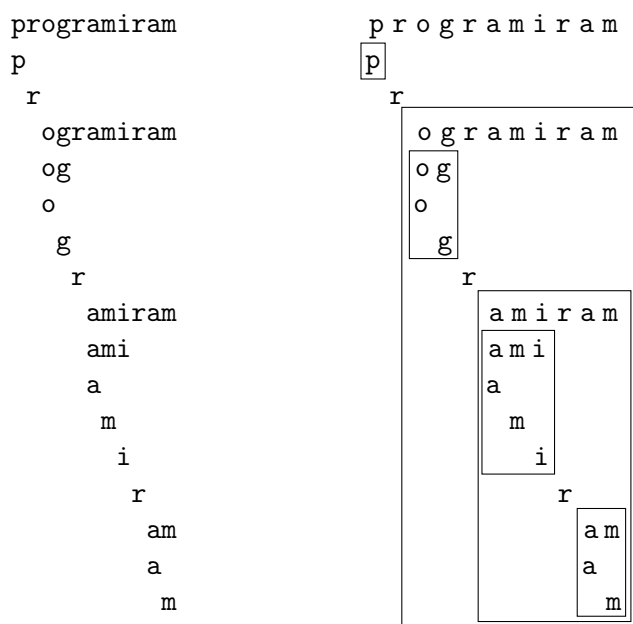
V vseh testnih primerih velja $n \in [3, 200]$, $m \in [3, 200]$ in $d \in [1, (\min(n, m) - 1) / 2]$. V 34% testnih primerov velja $n = 3$ in $d = 1$, v nadaljnjih 34% primerov pa $d = 1$.

- ③ Napišite program (`nalog3.c`), ki s standardnega vhoda prebere niz, sestavljen iz največ 200 alfanumeričnih ASCII-znakov, na standardnem izhodu pa proizvede izpis v skladu s sledečimi pravili:

- Če je niz prazen, naj program ne izpiše ničesar. (Vhodni niz bo sicer neprazen, lahko pa bodo prazni nekateri od nizov, ki bodo nastali v postopku.)
- Če niz vsebuje en sam znak, naj program izpiše samo ta znak (na ustreznem zamiku, kot bo razvidno iz nadaljnjega besedila).
- Če je niz enak $c_0 c_1 \dots c_{n-1}$ za nek $n \geq 2$ in če je m indeks tistega znaka v nizu, ki ima največjo ASCII-kodo (če je več takih znakov, je m indeks prvega med njimi), potem naj program tvori izpis takole:



Na primer, niz `programiram` razbijemo na niz `p`, znak `r` in niz `ogramiram`. Niz `p` samo izpišemo, niz `ogramiram` pa razbijemo na niz `og`, znak `r` in niz `amiram`. Niz `og` razbijemo na prazen niz, znak `o` in niz `g`, niz `amiram` pa na niz `ami` (\Rightarrow a, m, i), znak `r` in niz `am` (\Rightarrow a, m, prazen niz). Leva slika prikazuje pripadajoči izpis:



V 30% testnih primerov so znaki v nizu urejeni po padajoči ASCII-kodi. Če se pri nekem testnem primeru zmotite samo pri izpisu vodilnih presledkov (zamikov) ali pa če zamike preprosto ignorirate, boste za tisti primer prejeli 50% točk.