

## opisi odgovorov na moje naloge

### 1 - Križci in krožci

Podprogram Izenaceno hodi po nizu enak po znak in si hrani relativno prevlado xa ali oja v spremenljivki. Če se pojavi x, poveča spremenljivko, če se pojavi y, jo zmanjša. Tako bo spremenljivka ob izenačenem izidu 0. Če zazna znak, ki ni bodisi x bodisi o, nemudoma odvrne, da ni izenačeno. Ob predpostavljajanju, da sta bila pregledana že vsaj dva znaka, primerja vrednost tega, prejšnjega in predprejšnjega znaka, če so vsi enaki, izid ni izenačen.

Vstopna funkcija programa samo poda prvi argument v funkcijo Izenaceno.

### 2 - Kovanci

podprogram prejme tabelo s kovanci in njeno dolžino ter v zanki iterira čez vsak kupček. V kolikor je trenutni kupček večji od naslednjega, kupček pobere - beri: poveča spremenljivko s seštevkom - ter pobere vsak drugi kupček od tega proti začetku seznama kupčkov.

Vstopna funkcija programa samo pobere argumente in jih po vrsti pretvori v cela števila ter nato seznam in število kupčkov poda v funkcijo podprogram.

### 3 - Taksi

podprogram dobi seznam stalnih strank in seznam možnih central ter njuni velikosti, nato pa v zanki iterira skozi vse možne centrale ter za vsako centralo iterira po stalnih strankah ter s podano funkcijo sešteva njihove poti do stranke. V kolikor program najde centralo, ki ima manjši seštevke poti od prejšnje, jo zabeleži, ob koncu zanke torej odvrne centralo z najkrajšim seštevkom poti do stalnih strank.

Vstopna funkcija programa samo prebere standardni vhod in od tam pridobi možne centrale ter stalne stranke, jih pretvori v seznama, ki jih poda podprogramu, ter izpiše najugodnejšo centralo.

### 4 - Preusmerjanje

podprogram dobi seznam preusmeritev in njegovo velikost ter vstopno točko oziroma začetni naslov. Nato v zanki iterira skozi vse naslove na seznamu in, ko najde preusmeritev, najprej preveri, da je ni že obiskal - če jo je, odvrne, da je zaciklan, jo označi kot obiskano, ter pokliče in odvrne samega sebe z istimi parametri, le zamenja začetni naslov na naslov, kamor ga preusmeritev preusmerja. Če v zanki ne najde preusmeritve, odvrne začetni naslov.

Vstopna funkcija programa samo prebere standardni vhod in od tam pridobi seznam preusmeritev, kot argument pa dobi začetni naslov. Nato jih pretvori v cela števila ter pokliče podprogram in ustrezno v standardni izhod zapiše ugotovitev.

## 5 - Odstranjevanje črk

podprogram dobi seznam nizov in njegovo dolžino ter niz iz prejšnjega rekurzičnega klica. Najprej v zanki najde nize, ki imajo enako velikost, kot prvi niz v seznamu ter v zanki iterira čez njih. Pri vsakem v zanki iterira skozi njegove črke in jih primerja s črkami prejšnjega niza, če je potrebno izpustiti zgolj eno črko niz označi kot aplikabilen in odvrne rekurziven klic nase, ki kot prejšnji niz navede ta niz in kot seznam ta seznam, le zamaknjen tako, da ne vsebuje že obdelane dolžine. Pred rekurzivnim klicem seveda preveri, da nismo slučajno že v največji dolžini niza, v tem primeru takoj po spoznanju, da je niz aplikabilen, tega takoj odvrne. Če se glavna zanka konča - beri: pregledamo vse nize trenutne velikosti - odvrne prejšnji niz in s tem zaključí rekurzijo.

podprogram ne neha iskati nizov po prvem najdenem nizu, temveč nadaljuje in odvrne tak niz, pri katerem rekurziven klic vrne najdaljši niz.

Niz je definiran kot seznam bajtov in njihova dolžina, da sortirnemu algoritmu ni treba vedno klicati `strlen()`.

Vstopna funkcija programa po velikosti uredi argumente ter jih od kratkih proti daljšim pošlje v podprogram ter napiše ugotovitev podprograma v standardni izhod.