

Opisi rješenja 7. razred

Kupovina

Potrebno znanje: osnovne naredbe za kretanje kornjače, koordinatna grafika, liste, crtanje kružnice, osnovne operacije za kontrolu tijeka programa (REPEAT, FOR)

Počinjemo crtati od lijevog kotača i krećemo se prateći skicu po lijevoj strani kolica. Umjesto da se krećemo naprijed za :a u jednom potezu, koristimo naredbu REPEAT kojom :n puta crtamo kratke segmente duljine :a / :n i nakon crtanja svaki put zapamtimo po jednu točku u listu :lijevo koju ćemo trebati kasnije. Sada pamtimu točku gdje počinje ručka jer će nam trebati kasnije za spajanje gornjeg ruba košare kolica. Zatim crtamo ručku prema skici i vraćamo se natrag.

Sada se krećemo donjom stranom kolica i pamtimu točke na dnu. Opet koristimo repeat kako bismo prelazili kraće duljine, ovaj put duljine :a / :m. Točke koje sada pamtimu spremamo u listu :dolje.

Na kraju donje stranice crtamo kotač prema skici i zatim crtamo desni rub kolica. Sad nam je korisna naredba FOR kojom ponavljamo crtanje segmenata duljine :a / :n i istodobno pratimo točke koje smo prethodno spremili u listu. Nakon što nacrtamo pojedini segment pamtimu točku do koje smo došli u varijablu :desna_točka te točku spajamo s odgovarajućom točkom iz liste :lijevo s kojom je točka u paru.

Prilikom crtanja gornje stranice kolica koristimo sličnu strategiju u kojoj se krećemo po crti u više koraka, ali ovaj put treba paziti da se listu treba obilaziti od kraja k početku dok crtamo jer smo se prilikom crtanja donje crte kretali s lijeva na desno, a sad se krećemo s desna na lijevo. Također, potrebno je izračunati duljinu crte na gornjem rubu kolica. To možemo pomoću naredbe distance jer smo prethodno zapamtili točku na suprotnoj strani.

Za ostvarivanje 16,67% (5) bodova nije potrebna koordinatna grafika ni liste jer vrijedi :k = 90 pa je oblik košare kolica biti pravokutan, te vrijedi :n = 1 i :m = 1, što znači da nema žica koje dijele kolica ni okomito ni vodoravno.

Za ostvarivanje dodatnih 16,67% (5) bodova potrebno je crtati samo poprečne crte jer vrijedi :m = 1. Potrebno je korištenje koordinatne grafike, ali je dovoljna jedna lista.

Za ostvarivanje dodatnih 16,67% (5) bodova nije potrebna koordinatna grafika ni liste jer je oblik košare pravokutan (vrijedi :k = 90), ali treba nacrtati

Drugi dio zadatka bio je potražiti zadanu riječ :w. Pomoću dvije FOR petlje koje se nalaze jedna u drugoj moglo se isprobati svaku kombinaciju za redak :i i stupac :j kao potencijalni početak riječi. Za svaku od mogućih početnih pozicija bilo je potrebno provjeriti nalazi li se tražena riječ u jednom od mogućih 8 smjerova. Ako je na početnoj poziciji već u početku pogrešno slovo tu povodoravne i uspravne crte.

Keks

Potrebno znanje: osnovne naredbe za kretanje kornjače, crtanje mnogokuta, osnove koordinatne grafike (naredbe SETX, XCOR i DISTANCE), osnovne operacije za kontrolu tijeka programa (REPEAT, IF), crtanje kružnice i polukruga

Službeno rješenje najprije crta ugriz, a zatim čokoladne komadiće i obrub. Crtamo tako da je desni kraj ugriza od kojeg počinjemo crtati na poziciji (0,0). Ugriz se sastoji od polukružnica polumjera :b. Crtanje ugriza počinje od donje desne polukružnice koja ima polumjer :b, a crta se samo njezina gornja polovica, odnosno četvrtina kružnice polumjera :b. Pomoću naredbe REPEAT crtamo ostale polukružnice nad stranicama gornje polovice pravilnog :m - terokuta koji predstavljaju segmente ugriza. Crtanje završava četvrtinom polukružnice na lijevoj strani ugriza.

U ovom trenutku želimo saznati x koordinatu središta pravilnog :m - terokuta. Kako znamo da smo krenuli crtati ugriz s pozicije (0,0), a sad smo na suprotnoj strani ugriza, tražena vrijednost je polovina x koordinate na kojoj je trenutno nalazimo (dobivamo ju pomoću XCOR). Pomaknemo se na središte ugriza pomoću naredbe SETX i idemo za :a prema gore da bismo bili u središtu keksa čiji ćemo obrub nacrtati kasnije.

U središtu keksa crtamo središnji komadić čokolade polumjera :d. Dobivamo duljinu polumjera keksa, koji je udaljenost od trenutne točke na kojoj se nalazimo do točke ishodišta (0,0) naredbom DISTANCE. Moramo dovršiti crtanje komadića čokolade koji su kružnice raspoređene oko središnjeg komadića čokolade. Za kraj crtamo obrub keksa koji je kružnica polumjera koji smo ranije izračunali.

Za ostvarivanje 25% (10) bodova vrijedilo je :b = 0 pa nije bilo potrebno crtati ugriz. Kako je u tom slučaju ugriz beskonačno malen ponašamo se kao da smo se našli u njegovom središtu te rješavamo zadatak prateći službeno rješenje od dijela kad se krećemo naprijed za :a prema središtu keksa.

Za ostvarivanje dodatnih 25% (10) bodova vrijedilo je :m = 6. U ovom slučaju je lakše izračunati kuteve potrebne za crtanje ugriza te nam nije potrebna koordinatna grafika da bismo saznali gdje se nalazi središte ugriza.

Osmosmjerka

Potrebno znanje: obilazak liste, rad s riječima, operacije za kontrolu tijeka programa (REPEAT, FOR), osnovne naredbe za kretanje kornjače, ispunjavanje omeđenog dijela ekrana bojom (SETF, FILL)

Prvi dio zadatka je nacrtati izgled osmosmjerke. U zadatku je garantirano da će sve zadane riječi biti jednake duljine pa po duljini prve riječi znamo koliko ima stupaca, a po broju riječi zadanih u listi :l znamo koliko ima redaka u osmosmjerki. Kad nam je poznat broj redaka i stupaca možemo nacrtati odgovarajuće kvadrate sa stranicom :a.

ziciju se preskače. U svakom smjeru ide se WHILE petljom dok se ne dogodi da izađemo iz osmosmjerke ili dok ne pregledamo onoliko slova koliko ima riječ :w. Cijelo vrijeme brojimo koliko je slova u trenutno pregledanom smjeru bilo ispravno. Ako je na kraju provjere broj isprvnih slova bio isti kao i u riječi :w to znaci da smo pronašli sva slova te se odgovarajuća polja osmosmjerke boje u crno. Rješavanje je lakše ukoliko se koristimo procedurama za pregledniju organizaciju koda.

U testnim primjerima vrijednim ukupno 10% (6) bodova, tražena riječ :w nije bila sadržana u osmosmjerci pa ju nije trebalo obojiti i bilo je dovoljno samo nacrtati praznu osmosmjerku, odnosno riješiti prvi dio zadatka.

U testnim primjerima vrijednim ukupno 10% (6) bodova, tražena riječ :w sadržavala je samo jedno slovo pa nije trebalo raditi provjeru u 8 smjerova i bilo je dovoljno samo provjeriti sv epočetne pozicije.

U testnim primjerima vrijednim ukupno 20% (12) bodova, tražena riječ :w pojavila se postavljena vodoravno i imala je više od jednog slova. Za ostvarivanje tih bodova bilo je dovoljno provjeriti samo dva smjera: vodoravno s lijeva na desno i vodoravno s desna na lijevo.

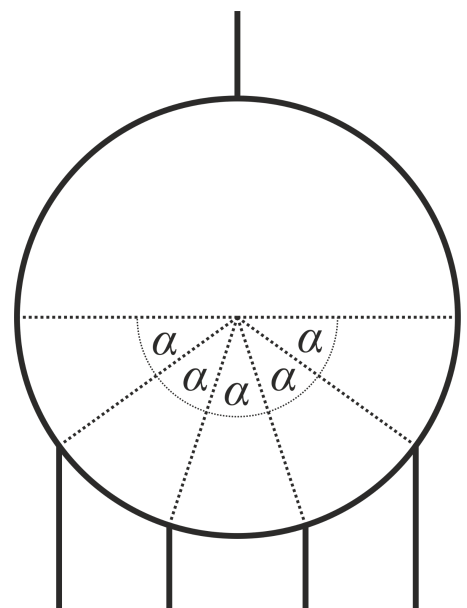
U testnim primjerima vrijednim ukupno 20% (12) bodova, tražena riječ :w pojavila se postavljena okomito i imala je više od jednog slova. Za ostvarivanje tih bodova bilo je dovoljno provjeriti samo dva smjera: okomito odozgo prema dolje i okomito odozdo prema gore.

Dreamcatcher

Potrebno znanje: koordinatna grafika, rekurzija, operacije za kontrolu tijeka programa (REPEAT, FOR), crtanje kružnice

Počnemo rješavanje tako da namjestimo kornjaču na zadanu poziciju [0 400]. Zbog te naredbe glavna procedura DREAMCATCHER :r :d :l ne može biti rekurzija. Kad bi bila rekurzija svakim pozivom bi se vraćala na početnu poziciju. Rekurzivni dio zadatka obavlja se u pomoćnoj proceduri nazvanoj DCREK :r :d :l u službenom rješenju. Procedura DCREK poziva se iz glavne procedure nakon pomaka na poziciju.

U rekurziji DCREK na početku pamtimo usmjerenje i položaj kornjače kako bismo se na kraju poziva rekurzije lakše vratili na početni položaj i usmjerenje. Usmjerimo kornjaču prema dolje te crtamo uže i kružnicu polumjera :r. Iz trenutne liste :l uzimamo prvi broj koji označava broj manjih kružnica nazvan :n.



Manje kružnice možemo nacrtati pomoću naredbe REPEAT koja ponavlja naredbu CIRCLE2 (crta kružnicu pomičući kornjaču rubom kružnice) te okreće kornjaču za $360 / :n$ stupnjeva kako bi sljedeće kružnica počinjala s pomaknutim usmjerenjem.

Broj privjesaka $:m$ dobivamo tako da od duljine liste oduzmemo 1. Varijablu $:alfa$ koja prema skici određuje ramak između privjesaka dobivamo kao $180/(:m+1)$. Ako je $:m = 0$ vraćamo se na početnu poziciju i usmjerenje te završavamo rekurziju. U suprotnom FOR petljom obilazimo listu te pozivamo rekurziju za svaki podelementkoji je lista. Za elemente koji nisu liste pozivamo pomoćnu proceduru PERO $:d$ koja crta pero prema skici danoj u zadatku. Nakon obilaska liste vraćamo se na zapamćeni položaj i usmjerenje.

Za osvajanje ukupno 28% (20) bodova nije bilo potrebno napisati rekurzivno rješenje jer je bilo obećano da lista $:l$ neće imati podliste, što znači da je dreamcatcher mogao biti bez privjesaka ili su svi privjesci bili pera.

Za osvajanje ukupno 57% (40) bodova bilo je potrebno napisati malo naprednije, ali i dalje nerekurzivno rješenje jer je bilo garantirano da lista $:l$ neće imati podliste s više od jednog člana pa su privjesci na dreamcatcheru bili ili pera ili dreamcatcher s perima.