

Upute za predaju rješenja:

- Rješenje se predaje e-mailom na adresu **svog** asistenta
- Rješenje predano nakon gornjeg roka neće se uvažiti
- Rješenje treba biti zapakirano u arhivu naziva **SPA- Z02-ImePrezime**

Zadatak 1: I4, 2 boda

U prilogu je aplikacija kakvu trebate napraviti, a sastoji se od sljedećeg: prilikom pokretanja aplikacije korisnik unosi redak i stupac početne točke A i redak i stupac krajnje točke B (broj retka ide od 1 do 20, a broj stupca od 1 do 40). Nakon toga, aplikacija treba osmisliti put od točke A do točke B i svakih 100 milisekundi iscrtati gdje se trenutno nalazimo. Točka A i točka B se mogu nalaziti na bilo kojem dopuštenom mjestu (isprobajte kako radi aplikacija u prilogu s, primjerice, A = 1, 1 i B = 20, 40, te kako radi s A = 20, 20 i B = 10, 10).

Zadatak 2: I3, 1 ekstra¹ bod i I4, 1 ekstra bod

Prethodna aplikacija je odlična u slučaju kad nema neprelaznih prepreka. Međutim, ako se na igralištu pojave zidovi, morat ćemo mijenjati način dolaska od A do B tako da ne prolazimo kroz zidove, već da ih zaobiđemo. Druga aplikacija demonstrira takav način rada (isprobajte kako radi aplikacija u prilogu s A = 1, 1 i B = 1, 40 – uredno ćemo zaobići zid i stići do cilja. Navedeni princip vrijedi koliko god zidova postavimo na igralištu).

Vaš zadatak je isprogramirati aplikaciju tako da se igrač na putu od A do B zaobilazi zidove i sretno stiže do odredišta. Postoji puno načina za to napraviti, a najjednostavniji je primjenom Dijkstrinog algoritma pronalaska najkraćeg puta:

- Opis i primjer rada algoritma na hrvatskom:
http://www.veleri.hr/files/datoteke/nastavni_materijali/k_promet_s1/Kvantitativne_4_Promet.pdf
- Odlična implementacija koju smijete koristiti (smijete i druge, naravno, a možete napisati i svoju, ali ova je najkraći put do rješenja): http://rosettacode.org/wiki/Dijkstra%27s_algorithm

¹ Ekstra bodovi se uračunavaju u bodove s međuispita po formuli: broj bodova na MI2 = MIN(osvojeno na MI2 + ekstra bodovi, 14), za svaki od ishoda.