

Zadatak 1:

1. Komponente dizajna:
 - Cilj je minimizirati duljinu prijednog puta u obilasku za svaki kriterij zasebno
 - Rješenje prikazano kao lista s permutacijama 'gradova'
 - Početno rješenje svake instance je pohlepno rješenje iz prošle zadaće
 - 2-opt – zamijeni dva brida ako tom zamjenom dobivamo bolji makespan odnosno traveltime, ovisno o kriteriju
 - TABU lista – promjena prethodno promijenjenih bridova zabranjena je 2 iteracije
 - Matrica susjedstva
2. Implementacija algoritma u „dz2_zad1.py“
3. Načini za poboljšanje implementiranog algoritma
 - Ne pretraživati cijelo susjedstvo nego samo dio -> znatno poboljšanje u brzini pretraživanja ali rezultati gotovo isti
 - Rezultati spremljeni u datoteke: 'dz2_fixed_tabu_makespan.txt', 'dz2_fixed_tabu_makespan_solutions.txt' za makespan kriterij, odnosno u 'dz2_fixed_tabu_traveltime.txt' i 'dz2_fixed_tabu_traveltime_solutions.txt' za traveltime kriterij
 - Isprobati 3-opt ili switch-one metodu

Zadatak 2:

1. Komponente dizajna:
 - Cilj je minimizirati duljinu prijednog puta u obilasku za svaki kriterij zasebno
 - Rješenje prikazano kao lista s permutacijama 'gradova'
 - Funkciji za SA proslijeđuje se odgovarajuća csv datoteka da bi dobili sve instance za kriterij, početna temperatura, temperatura pri kojoj se zaustavlja i broj iteracija na svakoj temperaturi
 - U svakoj iteraciji susjed se odabire nasumično
 - Provjera za svaki od kriterija radi se pri provjeri je li susjed poboljšavajući ili ne
2. Implementacija algoritma u „dz2_zad2.py“
 - Rezultati spremljeni u datoteke: 'dz2_SA_traveltime_solutions.txt' i 'dz2_SA_makespan_solutions.txt'