



U direktorijumu **Rad** koji se nalazi na desktopu kreirati direktorijum **Ime\_prezime\_indeks** i u okviru njega kreirati fajl **Licitacija.c** koji sadrži program koji rešava problem korišćenjem dinamičkih struktura podataka stabla (Crveno-Crna maksimum poena je 20, ukoliko se zadatak reši korišćenjem AVL stabla maksimum je 17 poena). Program treba da organizuje raspodelu novogodišnjih poklona. Na Severnom polu se nalazi **n** Deda Mrazova, i svaki od njih ima svoj jedinstven ID (ceo broj), koordinate **X** i **Y** (realni brojevi, koji predstavljaju lokaciju njegovih poklona), broj dece (ceo broj) i niz njemu dodeljene dece (dete se pamti preko svog jedinstvenog ID-a, koji je ceo broj – članovi tog niza su jedinstveni). U svrhu lakše organizacije koristi se struktura podataka pogodna za bržu pretragu, tj. u ovom slučaju binarno pretraživačko stablo.

Stablo se uređuje prema udaljenosti Deda Mraza u odnosu na Severni pol, za koji se pretpostavlja da se nalazi na koordinati (0,0).

- ❖ Nakon završenog rasporeda štampati podatke o Deda Mrazovima od najmanje udaljenosti ka najvećoj.
- ❖ Napisati rekurzivnu funkciju koja **za dati koren stabla** pronalazi Deda Mraza koji je najbliži severnom polu a ima najviše dece (ukoliko ima više njih na istoj udaljenosti i sa istim brojem dece, vratiti prvog od njih)
- ❖ U glavnom delu programa štampati podatke o prethodno pronađenom Deda Mrazu.
- ❖ Došlo je do izvesne greške u organizaciji i neka deca se nalaze na više različitih spiskova. Kako ne bi dobila dva ili više poklona, potrebno ih je ostaviti samo na spisku kod jednog Deda Mraza (izabrati bilo kog). Ono što može da se desi je da neki Deda Mraz ostane bez dodeljene dece, u tom slučaju potrebno ga je izbaciti iz stabla. (Brisanje bez kreiranja novog stabla donosi bonus od 3 poena.)
- ❖ Nakon završenog ažuriranja štampati podatke o Deda Mrazovima od najmanje udaljenosti ka najvećoj.