

Napisati:

1. klasu *Mesto* koja čuva podatke o mestu - naziv (koji ne sadrži beline) i koordinate mesta. Obezbediti konstruktor koji kao argumente ima naziv mesta i njegove koordinate, metode koji vraćaju koordinate mesta, metod *bool dostupna(final Antena a)* koji vraća da li je mesto pokriveno antenom *a* i metod *String toString()* - format zapisa mesta je naziv, prazno mesto, zatim koordinate mesta razdvojene jednom praznim mestom.
2. klasu *Antena* koja čuva podatke o anteni - koordinate antene i *A* i *B* - čije značenje zavisi od tipa antene. Ova klasa je apstraktna i predstavlja baznu klasu za razne tipove antena. U njoj obezbediti konstruktor, metod *bool uDometu(int x, int y)* koji vraća da li je tačka sa koordinatama (x, y) u dometu antene, metod *char naziv()* koji vraća oznaku antene (P za pravougaonu, L za linearnu), metode koji vraćaju koordinate antene, metod *bool uDometu(final Mesto m)* koji vraća da li je mesto *m* u dometu antene, metod za upoređivanje da li sa antena poklapa sa datom (za dve antene smatrao da su iste ako su iste vrste, imaju iste koordinate i domet) i metod *String toString()* - format zapisa antene je je oznaka antene, prazno mesto, zatim koordinate antene odvojene jednim praznim mestom, prazno mesto i zatim *A* i *B* odvojeni jednim praznim mestom.
3. klase *PAntena* i *LAntena* kao potklase klase *Antena*. U svakoj od njih obezbediti konstruktor koji kao argumente ima koordinate antene i domet i obezbediti pravilno funkcionisanje svih metoda.
4. klasu *Država* koja čuva podatke o svim mestima i svim antenama u jednoj državi kao i konstante koje predstavljaju koordinate donjeg levog i gornjeg desnog ugla države i konstantu MAX. Obezbediti konstruktor na osnovu zadatih konstanti (skup mesta i antena je na početku prazan - realizovati ih preko klase *Vector*), metod *bool dodajAntenu(Antena a)* koji dodaje novu antenu i vraća da li je dodavanje uspešno obavljeno (antena se ne dodaje ako postoji barem jedno mesto koje bi time bio pokriveno sa više od MAX antena), metod *bool ukiniAntenu(int x, int y)* koji uklanja antenu koja se nalazi na koordinatama (x, y) i vraća da li je uklanjanje uspešno obavljeno (smatramo da je neuspešno ako na tim koordinatama nema antene), metod *void dodajMesto(final Mesto m)* koji dodaje novo mesto i metod *int pokrivenost()* koji vraća odnos broja mesta koja su pokrivena barem jednom antenom i broja tačaka u državi (rezultat je zaokružen na ceo broj).
5. glavni program koji iz datoteke "mesta.txt" učitava podatke o mestima, iz datoteke "antene.txt" učitava podatke o antenama, a iz datoteke "drzava.txt" podatke o državi pa na standardnom izlazu prikazuje pokrivenost teritorije i antene koje ne smeju da budu postavljene zbog preopterećenja.

Napomena: Napraviti projekat sa svojim imenom (primer: PetrovicPetar) i u njemu ostaviti sve fajlove. Neophodno je pisati komentare.

Napisati:

1. klasu *Mesto* koja čuva podatke o mestu - naziv (koji ne sadrži beline) i koordinate mesta. Obezbediti konstruktor koji kao argumente ima naziv mesta i njegove koordinate, metode koji vraćaju koordinate mesta, metod *bool dostupna(final Antena a)* koji vraća da li je mesto pokriveno antenom *a* i metod *String toString()* - format zapisa mesta je naziv, prazno mesto, zatim koordinate mesta razdvojene jednom praznim mestom.
2. klasu *Antena* koja čuva podatke o anteni - koordinate antene i *A* i *B* - čije značenje zavisi od tipa antene. Ova klasa je apstraktna i predstavlja baznu klasu za razne tipove antena. U njoj obezbediti konstruktor, metod *bool uDometu(int x, int y)* koji vraća da li je tačka sa koordinatama (x, y) u dometu antene, metod *char naziv()* koji vraća oznaku antene (P za pravougaonu, L za linearnu), metode koji vraćaju koordinate antene, metod *bool uDometu(final Mesto m)* koji vraća da li je mesto *m* u dometu antene, metod za upoređivanje da li sa antena poklapa sa datom (za dve antene smatrao da su iste ako su iste vrste, imaju iste koordinate i domet) i metod *String toString()* - format zapisa antene je je oznaka antene, prazno mesto, zatim koordinate antene odvojene jednim praznim mestom, prazno mesto i zatim *A* i *B* odvojeni jednim praznim mestom.
3. klase *PAntena* i *LAntena* kao potklase klase *Antena*. U svakoj od njih obezbediti konstruktor koji kao argumente ima koordinate antene i domet i obezbediti pravilno funkcionisanje svih metoda.
4. klasu *Država* koja čuva podatke o svim mestima i svim antenama u jednoj državi kao i konstante koje predstavljaju koordinate donjeg levog i gornjeg desnog ugla države i konstantu MAX. Obezbediti konstruktor na osnovu zadatih konstanti (skup mesta i antena je na početku prazan - realizovati ih preko klase *Vector*), metod *bool dodajAntenu(Antena a)* koji dodaje novu antenu i vraća da li je dodavanje uspešno obavljeno (antena se ne dodaje ako postoji barem jedno mesto koje bi time bio pokriveno sa više od MAX antena), metod *bool ukiniAntenu(int x, int y)* koji uklanja antenu koja se nalazi na koordinatama (x, y) i vraća da li je uklanjanje uspešno obavljeno (smatramo da je neuspešno ako na tim koordinatama nema antene), metod *void dodajMesto(final Mesto m)* koji dodaje novo mesto i metod *int pokrivenost()* koji vraća odnos broja mesta koja su pokrivena barem jednom antenom i broja tačaka u državi (rezultat je zaokružen na ceo broj).
5. glavni program koji iz datoteke "mesta.txt" učitava podatke o mestima, iz datoteke "antene.txt" učitava podatke o antenama, a iz datoteke "drzava.txt" podatke o državi pa na standardnom izlazu prikazuje pokrivenost teritorije i antene koje ne smeju da budu postavljene zbog preopterećenja.

Napomena: Napraviti projekat sa svojim imenom (primer: PetrovicPetar) i u njemu ostaviti sve fajlove. Neophodno je pisati komentare.