

Termin1 zadaci

1 ZADATAK

Kreirati novi projekat koji će imati naziv Termin1Zadaci.

2 ZADATAK

U projektu Termin1Zadaci napraviti novi paket zadatak1 i u tom paketu kreirati novu klasu Zadatak1. Unutar klase definisati novu enumeraciju SEMAFOR i napisati main metodu. Za ovu enumeraciju definisati da ima vrednosti CRVENO, ZUTO_CRVENO, ZELENO i ZUTO_ZELENO. Treba implementirati logiku koja će omogućiti prelaz stanja između ovih vrednosti. Ako je trenutna vrednost CRVENO prelazi se na ZUTO_CRVENO, ako je trenutna vrednost ZUTO_CRVENO prelazi se na ZELENO, ako je trenutna vrednost ZELENO prelazi se na ZUTO_ZELENO i ako je trenutna vrednost ZUTO_ZELENO prelazi se na CRVENO. U main metodi implementirati prelazak iz jedne vrednosti u drugu:

- a) Koristeći switch naredbu i taj deo koda izdvojiti u posebnom bloku naredbi
- b) Koristeći if else naredbu i taj deo koda izdvojiti u posebnom bloku naredbi

3 ZADATAK

U projektu Termin1Zadaci napraviti novi paket zadatak2. Definirati novu promenljivu n koja čuva celobrojnu pozitivnu vrednost. Za implementaciju sledećih funkcionalnosti koristiti promenljivu n:

- a) Napisati for petlju koja prolazi kroz sve celobrojne vrednosti od 0 do vrednosti koju čuva promenljiva n i računa ukupan zbir svih tih vrednosti. Za ovaj kod u paketu zadatak2 napraviti posebnu klasu Zadatak2Verzija1 i u njoj main metodu u kojoj ćete realizovati ovaj deo zadatka.
- b) Napisati for petlju koja u intervalu vrednosti od 1 do vrednosti koju čuva promenljiva n uzima samo vrednosti koje su neparne i dobija rezultat množenjem svih tih vrednosti. Za ovaj kod u paketu zadatak2 napraviti posebnu klasu Zadatak2Verzija2 i u njoj main metodu u kojoj ćete realizovati ovaj deo zadatka.
- c) Napisati for petlju koja u intervalu vrednosti od -n do n uzima sve celobrojne vrednosti i upisuje ih u novi niz. Koristeći for each petlju prikazati sve članove kreiranog niza na ekranu. Za ovaj kod u paketu zadatak2 napraviti posebnu klasu Zadatak2Verzija3 i u njoj main metodu u kojoj ćete realizovati ovaj deo zadatka.

4 ZADATAK

U projektu Termin1Zadaci napraviti novi paket zadatak3. Implementirati funkcionalnosti koje koriste matricu sa proizvoljnim brojem elemenata. Za proveru implementiranih funkcionalnosti možete kreirati matricu kao sa slike 4.1.

4	6	9	4	6
1	3	5	8	2
4	2	7	8	3
3	6	4	2	9
4	2	5	6	7

Slika 4.1: Matrica

Napisati sledeće funkcionalnosti:

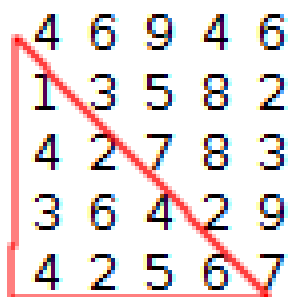
- Matricu prikazati na konzoli u istom formatu kao što je prikazana na slici 4.1. Između kolona treba da postoji prazno mesto, a svaki novi red matrice prikazuje se u novom redu na konzoli. Za ovaj kod u paketu zadatak3 napraviti posebnu klasu Zadatak3Verzija1 i u njoj main metodu u kojoj ćete realizovati ovaj deo zadatka.
- Napisati kod koji množi sve vrednosti sa glavne dijagonale. Na slici 4.2 su prikazani elementi sa glavne dijagonale u matrici koju možete koristiti za testiranje.

4	6	9	4	6
1	3	5	8	2
4	2	7	8	3
3	6	4	2	9
4	2	5	6	7

Slika 4.2: Elementi na glavnoj dijagonali

Za ovaj kod u paketu zadatak3 napraviti posebnu klasu Zadatak3Verzija2 i u njoj main metodu u kojoj ćete realizovati ovaj deo zadatka.

- Napisati kod koji množi sve elemente ispod glavne dijagonale. Na slici 4.3 su prikazani elementi koji se nalaze ispod glavne dijagonale matrice koju možete koristiti za testiranje.

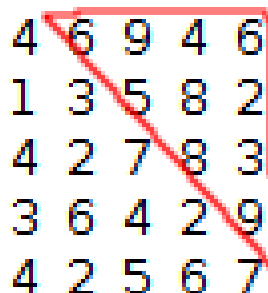


4	6	9	4	6
1	3	5	8	2
4	2	7	8	3
3	6	4	2	9
4	2	5	6	7

Slika 4.3: Elementi ispod glavne dijagonale

Za ovaj kod u paketu zadatak3 napraviti posebnu klasu Zadatak3Verzija3 i u njoj main metodu u kojoj ćete realizovati ovaj deo zadatka.

- d) Napisati kod koji pronalazi maksimalni element među elementima iznad glavne dijagonale. Na slici 4.4 su prikazani elementi koji se nalaze iznad glavne dijagonale matrice koju možete koristiti za testiranje.



4	6	9	4	6
1	3	5	8	2
4	2	7	8	3
3	6	4	2	9
4	2	5	6	7

Slika 4.4: Elementi iznad glavne dijagonale

Za ovaj kod u paketu zadatak3 napraviti posebnu klasu Zadatak3Verzija4 i u njoj main metodu u kojoj ćete realizovati ovaj deo zadatka.