

## Računarski praktikum 2 - drugi kolokvij, 27.05.2013.

Ime i prezime, username : \_\_\_\_\_ JMBAG: \_\_\_\_\_

Kolokvij se piše na računalu te traje dva sata. Zadatke spremate u datoteku

**ImePrezime\_broj.php** gdje broj označava broj zadatka koji rješavate. Svaku klasu potrebnu za rješavanje kolokvija spremate u **class\_ime.php** gdje ime označava ime klase. Na kolokviju je dozvoljeno korištenje materijala koji se nalaze na <http://demosi.math.hr/rp2/kolokvij2/> te vlastitih rješenja zadataka s vježbi koji se nalaze na vašem studentu. Kolokvij spremite u datoteku **kolokvij2.zip**, a tu datoteku spremite u vaš **public\_html** direktorij. Na datoteci postavite odgovarajuće ovlasti. Ako datoteku **nije moguće** dohvatiti, dobivate **nula** bodova na kolokviju.

**Podaci za spajanje na bazu su sljedeći:**

**username** = "rp2d", **password** = "rp2dw", **server** = "student.math.hr", **baza** = "rp2d"

**Zadatak 1.** [10 bodova] Napravite login formu koja prima username i password . Username i password provjerite u MySQL tablici *korisnici* (stupci se zovu *username* i *password*) . Ako su podaci valjani, prosljedite skriptu na "user.php" gdje će se ispisati njegovi podaci. Podatke dohvatite iz tablice *clanci* (**stupci**: *clanci\_id*, *username*, *naslov*, *tekst\_clanka*) te ih ispišite u tablicu.

Ispod tablice neka se nalazi drop down meni u kojem će biti ispisani svi stupci iz tablice. Odabirom stavke iz menija, tablica se mora sortirati po odabranom stupcu (za to možete koristiti JavaScript ili HTML formu). Dropdown meni neka se napuni tako da podatke dohvatite iz tablice (ako ne znate riješiti tako, unesite podatke ručno, ali tada zadatak nosi maksimalno 7 bodova).

Sve mora biti riješeno preko objekata (svaka klasa rješava samo svoje područje) te napravite getere i setere za klase koje to trebaju.

**Zadatak 2.** [10 bodova] Nadogradite prvi zadatak tako da implementirate sučelje *jačina* koje sadrži metode *vrati\_jacina* i *vrati\_ucestalost\_blokova*. Metoda *vrati\_jacina* neka vraća jačinu nekog teksta\*, a metoda *vrati\_ucestalost\_blokova* neka primi neki string od dva znaka, te neka vrati broj pojavljivanja tog stringa u tekstu (npr: ako je tekst “Ana voli Milovana” i metoda pozvana s parametrom “na”, metoda mora vratiti 2). Nadogradite i tablicu iz prvog zadatka tako da se uz ispis svakog članka ispisuje njegova jačina i učestalost bloka “rp”.

\*Kažemo da je članak jak ako sadrži više od 10 samoglasnika.

**Zadatak 3.** [5 bodova]

Napravite klasu *skalarni\_produkt* koja sadrži metodu *min\_scalar\_product*. Metoda prima dva niza (vektora) te traži permutaciju ta dvaju vektora koja će imati najmanji skalarni produkt koji će metoda vratiti kao parametar.

Primjer:

$$v_1 = [1, 3, -5]$$

$$v_2 = [-2, 4, 1]$$

Najmanji skalarni produkt od  $v_1$  i  $v_2$  je:  $-25$

Testne primjere i njihov najmanji skalarni produkt ispišite u tablicu sa stupcima “Vektor1”, “Vektor2” i “Najmanji skalarni produkt”.

Testirajte svoju metodu na sljedećim podacima:

a)

$$v_1 = [1, 2, 3, 4, 5]$$

$$v_2 = [1, 0, -2, -5, -8]$$

b)

$$v_1 = [4, -3, 8, 0]$$

$$v_2 = [1, 1, 2, -9]$$

**Zadatak 4.** [dodatih 5 bodova]

Svaki mobilni uređaj može slati i primiti poruke. Ako se vratimo malo u prošlost kada nisu postojali pametni telefoni, tipkovnice su sadržavale brojeve od 0-9 gdje je nula označavala razmak, a svako slovo se dobilo određenim ponavljanjem pritiska neke tipke. Ako bismo htjeli napisati dva puta isto slovo koje se nalazi na jednom broju, potrebno je pričekati neko vrijeme da se može opet pritisnuti tipka. U našem zadatku, čekanje će biti reprezentirano sa znakom razmaka.

Primjer:  $a = 2$ ,  $b = 22$ ,  $aa = 2\ 2$ ,  $bb = 22\ 22$ ,  $rp = 777\ 7$ ,  $na\ sta\ va = 662077778208882$

Napravite klasu *poruka* koja sadrži metode *primi\_poruku* i *pošalji\_poruku*.

Metoda *pošalji\_poruku* prima string kao parametar, te vraća numerički zapis tog stringa.

Metoda *primi\_poruku* prihvaća numerički zapis nekog stringa i vraća konvertirani string iz danog numeričkog zapisa.