

Programiranje (C) (NASTAVNIČKI SMJER) – drugi kolokvij, 29. lipnja 2007.

Ime i prezime: _____

JMBAG: _____

Zadaci	1	2	3	4	5	6	7	Σ
Bodovi	10	10	3	7	10	20	20	80
Rezultat								

Napomene: Sva rješenja napišite isključivo na papire sa zadacima, jer jedino njih predajete. Skice smijete raditi na drugim papirima koje će vam dati dežurni asistent. Ne zaboravite se potpisati na svim papirima koje predajete. Dozvoljeno je korištenje isključivo pribora za pisanje i brisanje. Kalkulatori, razne tablice, papiri, mobiteli i sl. nisu dozvoljeni!

Rezultati: ??? Možda i ranije na <http://degiorgi.math.hr/c/>

Vaše kolokvije možete vidjeti u ???

- [10] 1. Napišite funkciju koja kao argument prima kvadratnu matricu **a** (realnih brojeva) reda **n**, te vraća 1 ako je matrica gornje trokutasta (svi elementi različiti od nule nalaze se na dijagonali ili iznad nje) i na dijagonali ima strogo pozitivne brojeve; inače treba vratiti 0. Napišite i kako se funkcija poziva (ne treba cijeli program; samo primjer poziva).
- [10] 2. Napišite funkciju `double **napravi_sortiranu_matricu(double **mat, int a, int b);` koja kreira novu matricu (treba ju dinamički alocirati) dimenzija $a \times b$ dobivenu uzlaznim sortiranjem stupaca početne matrice **mat** (također dimenzija $a \times b$). Funkcija vraća pointer na novu matricu, a početna matrica treba ostati nepromijenjena.

- [3] 3. Bez korištenja funkcija iz `string.h` implementirajte funkciju

```
char* strcat(char *str1, const char *str2);
```

tako da ima istu funkcionalnost kao i istoimena funkcija iz standardne biblioteke. Povratna vrijednost funkcije je argument `str1`.

- [7] 4. Napišite funkciju koja kao argument prima polje stringova. Funkcija treba vratiti string iz tog polja u kojem se nalazi najviše velikih slova [A-Z]. Ukoliko ima više takvih stringova, funkcija treba vratiti onog koji se posljednji pojavljuje u polju. *Napomene: Duljinu stringova u polju smijete ograničiti na 1023. Zabranjeno je korištenje dodatnih nizova.*

- [10] 5. Napišite funkciju čiji je prototip `char* f(const char* s);` koja vraća novi string nastao iz `s` tako da se dupliciraju sva pojavljivanja znamenki u `s`. Npr. ako je ulazni string `"abc19X5"`, onda izlazni string treba biti `"abc1199X55"`. Za novi string alocirajte točno onoliko memorije koliko je potrebno. Obavezno provjeravajte da li je alokacija memorije uspješno provedena.

6.

- [3] (a) Definirajte novi tip podataka `broj` za pohranu cijelih brojeva u vezanoj listi (tako da bude moguće deklarirati varijablu kao `broj *b;`).
- [10] (b) Napišite funkciju koja prima pokazivač na početak vezane liste cijelih brojeva. Funkcija mora iz liste izbaciti najmanji i najveći parni broj (ako ima parnih brojeva u listi) i preko varijablinog argumenta vratiti te dvije vrijednosti. Ako parnih brojeva nije bilo u listi povratne vrijednosti su 0.
- [7] (c) Napišite odsječak glavnog programa u kojem se definiraju potrebne varijable i u kojem se poziva napisana funkcija.

- [20] 7. Napišite program koji iz datoteke `dugovi.in` učitava podatke o dugovima. Svaki redak te datoteke sadrži dva imena (riječi duljine manje ili jednake 20 znakova) te jedan pozitivan cijeli broj koji predstavlja iznos u kunama koji prva osoba duguje drugoj. Vaš program treba na temelju ove datoteke napraviti datoteku `dugovi.out` koja ima jedan redak u kojem piše ime osobe koja ukupno duguje najviše novca, te iznos koji ta osoba duguje. Dugovi se “prebijaju”, tj. ako Ana duguje Peri 100 kuna, a Pero duguje Anti 150 kuna, onda je Pero dužan ukupno 50 kuna. Možete pretpostaviti da u datoteci ima najviše 100 *različitih* imena.

Primjer:

dugovi.in	dugovi.out
Ana Pero 100	Ante 350
Pero Ante 150	
Ante Mirko 500	
Ana Mirko 100	