

MIKRORAČUNARI - ISPIT - SEPTEMBAR '11

1. (30 poena) Napisati *IA-32* asemblersku funkciju:

```
void digits(unsigned n, unsigned *c1, unsigned *c2);
```

koja određuje dve najveće cifre heksadekadnog zapisa celog broja $n \geq 0$. Najveću cifru smestiti na lokaciju na koju pokazuje *c1*, a drugu najveću cifru smestiti na lokaciju na koju pokazuje *c2*. Napisati potom i *C*-program koji sa standardnog ulaza učitava ceo broj n u heksadekadnom zapisu, zatim poziva funkciju i ispisuje njen rezultat (takođe u heksadekadnom zapisu) na standardni izlaz. Na primer, za ulaz:

```
0x5a3981
```

izlaz treba da bude:

```
a 9
```

2. (35 poena) Napisati asemblersku funkciju:

```
double log_rational(int m, double *p, int n, double *q, double x);
```

koja koristeći matematički koprocesor (*FPU*) izračunava vrednost funkcije:

$$R(x) = \frac{p_0 + p_1 \ln x + p_2 (\ln x)^2 + \dots + p_m (\ln x)^m}{q_0 + q_1 \ln x + q_2 (\ln x)^2 + \dots + q_n (\ln x)^n}$$

Funkcija prihvata redom stepen polinoma broioca m , pokazivač p na niz koeficijenata polinoma broioca počev od koeficijenta najmanje težine, stepen polinoma imenioca n , pokazivač q na niz koeficijenata polinoma imenioca (opet počev od koeficijenta najmanje težine) kao i vrednost $x > 0$ za koju se računa vrednost funkcije. Izračunata vrednost se vraća kao povratna vrednost funkcije. Napisati potom i *C*-program koji učitava ulazne podatke (u istom redosledu kao i u deklaraciji funkcije) i poziva funkciju, a zatim štampa rezultat na standardni izlaz. Prostor za nizove koeficijenata alocirati dinamički. Na primer, za ulaz:

```
2 2.1 3.2 1.2
3 0.3 1.1 5.1 4.2
3.4
```

izlaz treba da bude:

```
0.4600990639
```

3. (35 poena) Napisati *ARM* asemblersku funkciju:

```
void fractions(int f1[], int f2[], int r[]);
```

koja sabira dva razlomka. Svaki razlomak dat je nizom celih brojeva dužine 2, pri čemu je element na poziciji 0 brojilac, a element na poziciji 1 imenilac razlomka. Razlomci koji se sabiraju dati su nizovima $f1$ i $f2$, dok rezultujući razlomak treba smestiti u niz r . Dobijeni zbir treba da bude "skraćen", tj. njegovi brojilac i imenilac treba da budu uzajamno prosti. Napisati potom i *C*-program koji učitava razlomke koji se sabiraju (najpre brojilac i imenilac prvog razlomka, a zatim brojilac i imenilac drugog razlomka), poziva funkciju i ispisuje njen rezultat na standardnom izlazu. Na primer, za ulaz:

```
4 15
7 20
```

izlaz treba da bude:

```
37 60
```