

MIKRORAČUNARI - ISPIT - NOVEMBAR '10

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

```
int vowels(char * s);
```

koja izračunava broj uzastopnih parova samoglasnika u stringu na koji pokazuje pokazivač s . Pod samoglasnicima podrazumevamo karaktere 'a', 'e', 'i', 'o', 'u', 'A', 'E', 'I', 'O' i 'U'. Napisati potom i *C*-program koji učitava string sa standardnog ulaza, poziva funkciju i ispisuje njen rezultat na standardnom izlazu. Pretpostaviti da string neće biti duži od 1024 karaktera (sa sve terminirajućom nulom). Na primer, za ulaz:

```
emgeaegmewqiea
```

izlaz treba da bude:

```
4
```

2. (40 poena) Napisati *IA-32* asemblersku funkciju:

```
void complex_roots(double x, double y, int n, double *rx, double *ry);
```

koja, koristeći matematički koprocesor (*FPU*), izračunava sve kompleksne n -te korene datog kompleksnog broja $z = x + iy$. Kompleksni n -ti koreni z_0, z_1, \dots, z_{n-1} se računaju po sledećoj formuli:

$$z_k = \sqrt[n]{r} \left(\cos \left(\frac{\varphi + 2k\pi}{n} \right) + i \sin \left(\frac{\varphi + 2k\pi}{n} \right) \right) \quad k = 0, 1, \dots, n - 1$$

gde su $r = |z|$ i $\varphi = \arg(z)$. Realni deo korena z_k treba upisati u k -ti element niza na koji pokazuje pokazivač rx , dok imaginarni deo korena z_k treba upisati u k -ti element niza na koji pokazuje pokazivač ry (za svako $k \in \{0, 1, \dots, n - 1\}$). Napisati potom i *C*-program koji sa standardnog ulaza učitava x , y i n , poziva funkciju i ispisuje njen rezultat na standardnom izlazu. Na primer, za ulaz:

```
2.45 -1.23
```

```
7
```

izlaz treba da bude:

```
1.15241 -0.076712
0.778493 0.853163
-0.181647 1.14059
-1.005 0.569128
-1.07157 -0.430898
-0.331225 -1.10645
0.658542 -0.948821
```

3. (30 poena) Napisati *ARM* asemblersku funkciju:

```
int twobits(unsigned x);
```

koja izračunava broj uzastopnih parova jedinica u binarnom zapisu broja x . Napisati potom i *C*-program koji učitava vrednost broja x (kao neoznačeni heksadekadni broj), a zatim poziva funkciju i ispisuje njen rezultat na ekranu. Na primer, za ulaz:

```
0x1fab34ef
```

izlaz treba da bude:

```
12
```