

MIKRORAČUNARI - ISPIT - MART '11

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

```
int local_max(int * a, int n, int * b);
```

koja određuje sve lokalne maksimume u nizu dužine n na koji pokazuje pokazivač a . Za element $a[i]$ ($0 < i < n - 1$) kažemo da je lokalni maksimum ako važi $(a[i - 1] < a[i]) \wedge (a[i] > a[i + 1])$. Funkcija treba da u niz na koji pokazuje pokazivač b upiše redom sve indekse lokalnih maksimuma. Povratna vrednost funkcije je dužina kreiranog niza indeksa. Napisati potom i *C*-program koji sa standardnog ulaza učitava dužinu niza, alokira prostor za nizove, učitava elemente niza a , zatim poziva funkciju i ispisuje njen rezultat na standardnom izlazu. Na primer, za ulaz:

```
10
1 2 5 4 6 7 3 8 2 3
```

izlaz treba da bude:

```
2 5 7
```

2. (40 poena) Napisati asemblersku funkciju:

```
double variance(double * x, int n);
```

koja koristeći matematički koprocesor (*FPU*) izračunava disperziju slučajne veličine $Y = \frac{2^X}{X^4 + 1}$, gde je X diskretna slučajna veličina zadata nizom x koji sadrži n jednako verovatnih vrednosti. Disperzija se može izračunati formulom:

$$D(Y) = \frac{1}{n} \sum_{i=0}^{n-1} \left(\frac{2^{x_i}}{x_i^4 + 1} - E(Y) \right)^2$$

gde je $E(Y)$ očekivanje slučajne veličine Y koje se može izračunati formulom:

$$E(Y) = \frac{1}{n} \sum_{i=0}^{n-1} \frac{2^{x_i}}{x_i^4 + 1}$$

Napisati potom i *C* program koji učitava n kao i elemente niza x , poziva asemblersku funkciju i ispisuje rezultat. Na primer, za ulaz:

```
3
0.0 2.5 0.1
```

izlaz treba da bude:

```
0.178715
```

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

```
int twozeros(int x);
```

koja za dati broj x izračunava broj parova uzastopnih nula u njegovom binarnom zapisu. Na primer, broj `0xf09fafc8` se binarno zapisuje kao `1111000010011111101011111001000`, pa je broj parova uzastopnih nula 7. Napisati potom i *C*-program koji testira ovu funkciju. Ulaz se očekuje u heksadekadnom obliku. Na primer, za ulaz:

```
0xf09fafc8
```

izlaz treba da bude:

```
7
```