

MIKRORAČUNARI – ISPIT – NOVEMBAR 2012.

1. (35 poena) Napisati *IA-32* asemblersku funkciju:

```
int happy(int x);
```

koja ispituje da li je dati broj $x > 0$ srećan. Srećni brojevi se definišu na sledeći način: neka je data sekvenca brojeva x_0, x_1, x_2, \dots takva da je $x_0 = x$, a broj x_k je jednak zbiru kvadrata dekadnih cifara broja x_{k-1} , za $k > 0$. Ako se ova sekvenca završava brojem 1 (jer bi tada i svaki sledeći broj u sekvenci bio jednak 1), tada za početni broj x kažemo da je srećan. U suprotnom, ako se sekvenca ne završava (ne sadrži broj 1, već se periodično nastavlja u beskonačnost), tada broj x nije srećan. Može se dokazati da sekvenca brojeva koji nisu srećni uvek sadrže broj 4. Napisati potom i *C*-program koji sa standardnog ulaza učitava x , zatim poziva funkciju i ispisuje rezultat na standardnom izlazu. Na primer, za ulaz:

```
167
```

izlaz treba da bude:

```
1
```

2. (35 poena) Napisati *IA-32* asemblersku funkciju:

```
void diff(double * x, double * y, int n, double * r);
```

koja, koristeći paralelne *SSE2* instrukcije, za nizove dužine n na koje pokazuju pokazivači x i y izračunava sumu $\sum_0^{n-1} |x[i] - y[i]|$. Rezultat smestiti na lokaciju na koju pokazuje r . Napisati potom i *C*-program koji sa standardnog ulaza učitava n , alocira prostor za nizove, učitava elemente nizova, poziva funkciju i ispisuje njen rezultat na standardnom izlazu. Na primer, za ulaz:

```
3
```

```
0.45 1.2 -2.1
```

```
1.23 -2.31 1.24
```

izlaz treba da bude:

```
7.63
```

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

```
int compare(char *s, char *t);
```

koja leksikografski upoređuje dva data stringa. Funkcija vraća 1 ako je prvi string veći, -1 ako je manji, a 0 ako su jednaki. Napisati potom i *C*-program koji učitava dva stringa (čija dužina nije veća od 80 karaktera), poziva funkciju i ispisuje njen rezultat. Na primer, za ulaz:

```
abc123
```

```
abc132
```

izlaz treba da bude:

```
-1
```