

Uvod u organizaciju i arhitekturu računara 2

1 Asemblersko programiranje u ARM 32 arhitekturi - ZADACI

Pročitati izvod iz dokumentacije *arm.txt*. Pročitati *arm – uputstvo.txt* ili *arm – win – uputstvo.txt* i instalirati po uputstvu programe *qemu* i *armedslack – stripped*

1.1 Čas1 - uvod

1. **hello** - napisati program koji ispisuje na izlaz poruku.
2. **add1** - napisati funkciju *int add(int x, int y)* koja sabira dva broja.
3. **add2** - napisati funkciju *int add(int x, int y)* koja sabira dva broja.
4. **max1** - napisati funkciju *int max(int x, int y)* koja izračunava maksimum dva broja.
5. **max2** - napisati funkciju *int max(int x, int y)* koja izračunava maksimum dva broja.

1.2 Čas2 - petlje

6. **euler1** - napisati funkciju *int euler(int n)* koja odredjuje broj celih brojeva m , $1 \leq m \leq n$ takvih da su uzajamno prosti sa n .
7. **euler2** - napisati funkciju *int euler(int n)* koja odredjuje broj celih brojeva m , $1 \leq m \leq n$ takvih da su uzajamno prosti sa n .
8. **fact1** - napisati funkciju *int fact(int n)* koja izračunava faktorijel datog broja.
9. **fact2** - napisati funkciju *int fact(int n)* koja izračunava faktorijel datog broja.
10. **secret** - napisati funkciju *int secret(int x)* koja računa tajni broj datog broja.
11. **printbits** - napisati funkciju *void printbits(unsigned x)* koja prikazuje binarni zapis broja x na standardnom izlazu.
12. **mirror** - napisati funkciju *unsigned mirror(unsigned x)* koja vraća broj čiji je binarni zapis jednak binarnom zapisu broja x u ogledalu.

1.3 Čas3 - nizovi

13. **sum1** - napisati funkciju *int sum(int * a, int n)* koja izračunava sumu elemenata niza celih brojeva.
14. **sum2** - napisati funkciju *int sum(int * a, int n)* koja izračunava sumu elemenata niza celih brojeva.
15. **sum3** - napisati funkciju *int sum(int * a, int n)* koja izračunava sumu elemenata niza celih brojeva.
16. **max_element** - napisati funkciju *int max_element(int * a, int n)* koja odredjuje najveći element u nizu.

17. **minmax** - napisati funkciju *void minmax(int * a, int n, int * min, int * max)* koja određuje najveći i najmanji element niza celih brojeva.

1.4 Čas4 - nizovi vežbanje

18. **linsearch** - napisati funkciju *int linsearch(int * a, int n, int x)* koja ispituje da li se u nizu nalazi traženi element.

19. **quick_sort** - napisati funkciju *void quick_sort(int * a, int l, int r)* koja sortira niz celih brojeva u rastućem poretku.

20. **binsearch** - napisati funkciju *int binsearch(int * a, int n, int x)* koja ispituje da li se u rastuće sortiranom nizu nalazi traženi element.

21. **norms** - napisati funkciju *void norms(int ** a, int m, int n, int * h, int * v)* koja izračunava maksimum zbira apsolutnih vrednosti po vrstama, kao i maksimum zbira apsolutnih vrednosti po kolonama.

1.5 Čas5 - stringovi

22. **strlen** - napisati funkciju *int arm_strlen(char * s)* koja izračunava dužinu stringa *s*.

23. **palindrom** - napisati funkciju *int palindrom(char * s)* koja ispituje da li je dati string palindrom.

24. **most_frequent** - napisati funkciju *void most_frequent(char * s, char * c, int * f)* koja određuje najfrekventniji karakter u stringu *s*, kao i broj njegovih pojavljivanja.

25. **longest** - napisati funkciju *void longest(char * str, int * start, int * length)* koja pronalazi najdužu sekvencu istih karaktera u stringu.