

Dalyko „kompiuterių architektūra“ egzamino klausimai

2003 08 25

Variantas A

1. Registras AL = 08, registras AH = 9, registras CX = 0000. Kokia bus registro AX reikšmė šešioliktainėje sistemoje, įvykdžius komandą AAD?
2. AX = 0000, požymis AF = 1. Kokia bus registro AX reikšmė atlikus komandą AAA?
3. Užrašyti dešimtainį skaičių -17,17 slankaus kablelio formatu 4 baituose.
4. Įvykdžius nurodytą komandą, apskaičiuoti sekančios vykdomos komandos adresą, kai AX = 0003, BX = 0002, CX = 0000, DX = 0001:

FFFE EB FE jmp number (FFFE yra poslinkis kodo segmente)

5. Registrų reikšmės yra: DS = FF21, SS = 3456, CS = CC31, ES = E341, BP = 329A, BX = 3675, SI = FA45, DI = F122. Apskaičiuoti operando efektyvų adresą pagal adresavimo baitą AE.

Po adresavimo baito seka baitai: ABBA

6. Registrų reikšmės yra: DS = FF21, SS = 3456, CS = CC31, ES = E341, BP = 329A, BX = 3675, SI = FA45, DI = F122. Apskaičiuoti operando absoliutų adresą pagal adresavimo baitą AE.

Po adresavimo baito seka baitai: ABBA

7. Parašyti mikrokomandą MPL kalba, kuri užrašo reikšmę -2 į registrą MBR nenaudojant konstantinių registrų.
8. Registras SF = 0000. Baitų sudėties komanda prie dešimtainės reikšmės 200 pridedama dešimtainė reikšmė -100. Užrašyti naują registro SF reikšmę.
9. Registras SS = ABCD, registras SP = 0002, registras BP = AF00, registras CX = 0010. Kokia bus steko viršūnės reikšmė šešioliktainėje sistemoje, įvykdžius komandą:

3410 9A EBF1234 call text(3410 yra poslinkis kodo segmente)

10. Registrų reikšmės yra SI = EE8B ir DI = 12CD, registras CX = 0029, registras SF = 0000. Kokia bus registrų SI ir DI reikšmių suma, įvykdžius komandą rep lodsw?

Ats.:

1. 0062
2. 0106
3. C1 89 5C 29
4. FFFE
5. ED45

6. 432A5

7. $MBR = \text{LEFT_SHIFT}(MBR + \text{COM}(MBR))$

8. 0811

9. FFFE (alternatyvus pateiktas ats. 1534)

10. 01AA

Sprendimai:

1. Registras AX(AH;AL) = 0908. Pagal AAD veikimo algoritmą $AL = AH \cdot 10 + AL$ ir $AH := 00$.
 $AH \cdot 10 = 9 \cdot 10 = 90 = 5Ah$. $5Ah + 8 = 62$. Taigi AX = 0062.

2. AAA algoritmas:

if (((AL and 0Fh) > 9) or (AF = 1)) **then**

AL := AL + 6

AH := AH + 1

AF := 1

CF := 1

Else

AF := 0

CF := 0

Endif

AL := AL and 0Fh

(AL and 0Fh) sąlyga netenkinama, bet AF = 1, tad vykdoma **then** dalis. Po jos įvykdymo AX = 0106. Tuomet vykdoma AL := AL and 0Fh dalis, kuri šiuo atveju nieko nekeičia.

3. Pirma užsirašome skaičių dvejetainė skaičiavimo sistema. Pradedame nuo sveikosios dalies (į ženklą dėmesio nekreipiame) - $17_{10} = 10001_2$. Trupmeninė dalis $0,17_{10} = 00(10101110000101000111)$. Perrašome visą skaičių:

-10001,00(10101110000101000111). Skaičiaus eilė yra 4, tad charakteristika $4 + 7Fh = 83$. Slankaus kablelio formatu skaičius 4 baituose 31 bitas skirtas ženklo bitui, 30-23 charakteristikai, 23-0 mantisei. (skaičiuojama iš dešinės į kairę nuo 0). Šiuo atveju atitinkamai: **1100 0001 1000 1001 0101 1100 0010 1001**

1 - vienetasis, nes skaičius neigiamas

100 0001 1 - charakteristika 83 dvejetainėje skaičiavimo sistemoje.

000 1001 0101 1100 0010 1001 - mantisė, kurią sudaro skaitmenys dvejetainiame skaičiuje einantys po kablelio, jį perkėlus per eilę (šiuo atveju 4 skaitmenimis į priekį). Reiktų atkreipti dėmesį, jog šiuo atveju paskutinis skaitmuo yra 1, o ne 0 (koks būtų tiesiog nurašant), nes po jo eina 1 ir vyksta apvalinimas (10**10001**11 - trupmeninės dalies fragmentas).

Skaičių perrašius šešioliktainiu formatu gauname C1 89 5C 29.

4. EB komanda yra vidinis artimas valdymo perdavimas. Valdymas bus perduodamas **IP := IP + poslinkis**, kuris saugomas po EB einančiame baite. Šiuo atveju tai FE. Kadangi adresą sudaro 2 baitai, taigi plečiame FE pagal ženklo bitą. FE = 1111 1110 ženklo bitas 1, taigi

baitą užpildome 1 (1111 1111 1111 1110 = FFFE). IP yra FFFE (įvardintas, kaip poslinkis kodo segmente), bet skaitant komandą, kuri užima du baitus (EB ir FE) jis padidėjo komandos ilgiu. Galiausiai $FFFE + 2 = 0000$. Tuomet $IP + FFFE = FFFE$.

5. $AE = 1010\ 1110$. Pirmi 2 bitai yra mod laukas, tolesni 3 – reg, likę 3 – r/m. Šiuo atveju $mod = 10$, $r/m = 110$. Pagal **r/m** nustatome, jog efektyvus adresas bus formuojamas $BP +$ poslinkis. Pagal **mod** lauką nustatome, jog poslinkis 2 baitų. Poslinkyje ABBA baitai išsidėstę atvirkščia tvarka, t.y. pirma parašytas jaunesnysis baitas. Formuodami adresą juos sukeisime vietomis.
- $$EA = 329A + BAAB = ED45$$

6. $AA = seg.reg.*10h + EA$. Kadangi duomenys identiški 5 užduoties duomenims EA jau esame apskaičiavę. Kadangi formuodami EA naudojame BP registrą, tai segmento registras bus SS.

$$AA = 3456*10h + ED45 = 34560 + ED45 = 432A5$$

7. Įsimintina, jog $MBR + COM(MBR) = -1$. Panaudoję poslinkį į kairę LEFT_SHIFT, atliekame daugybą iš dviejų. Gauname -2. $MBR = LEFT_SHIFT(MBR + COM(MBR))$.

8. Pirmiausia perrašome skaičius į dvejetainę sistemą. $200_{10} = 1100\ 1000$, $-100_{10} = 1001\ 1100$. Tuomet sudedame reikšmes ir formuojame požymius: $CF = 1$, $PF = 1$, $AF = 1$, $ZF = 1$, $SF = 1$, $OF = 1$. Įrašę jas į atitinkamas SF vietas

(0000 1000 0001 0001) parašome reikšmę šešioliktainėje skaičiavimo sistemoje – 0881.

9. 9A – išorinio tiesioginio paprogramės iškvietimo komanda. Po jos eina 4 baitai – 1-2 jaunesnysis ir vyresnysis adreso baitai, 3-4 jaunesnysis ir vyresnysis segmento baitai. Komandos metu vykdomi tokie veiksmai:

PUSH CS

PUSH IP

CS := segmentas

IP := adresas

Įdedant CS ir IP į steką kas kartą steko viršūnės rodyklė SP mažinama 2. Pirmąjį kartą sumažinus rodyklę

($SP := 0002 - 0002 = 0000$) gauname didžiausią galimą steko dydį. Antrą kartą mažinant rodyklę nuo didžiausios galimos steko reikšmės 0000 ($SP := 0000 - 0002 = FFFE$) mažiname nuo maksimalios reikšmės FFFF. Galutinis atsakymas $SP = FFFE$.

10. Prefiksas **rep** kartoja komandą tol, kol $CX \neq 0$, kiekvieną kartą CX sumažindamas 1. Komanda **lods** akumuliatoriui (AX arba AL) priskiriama DS segmento reikšmė, į kurią rodo SI ir SI padidinamas/sumažinamas akumuliatoriaus ilgiu. Ar bus didinama/mažinama nustatome pagal DF bitą, esantį SF registre (0 – didinama, 1 – mažinama). SF visi bitai 0, tad bus didinama. Komandoje raidė **w** rodo, jog bus operuojama žodžio ilgio duomenimis, naudojamas AX registras, o SI didinamas/mažinamas 2. Komanda bus kartojama 29h kartus ($CX = 0029$). Kadangi kaskart didinsime 2, tai iš viso SI registras padidės 52h ($29h * 2$). $SI := EE8B + 52 = EEDD$, o DI registras šiuo atveju nekinta. Jų abiejų suma $SI + DI = EEDD$

+ 12CD = 101AA. Paprastai rašome 2 baitų atsakymą (01AA), bet šiuo atveju, matyt, tiktu ir 101AA.