CSC 3210

Computer Organization and Programming

Lab 4

Answer Sheet

Student Name: **Hemant Kosaraju**

Section: **020**

Debug through each line of code and explain the register content.

Line number: **8**

Instruction: **mov al, 245**

Register value: **EAX = 00B7FAF5**

Explanation: **The EAX register is filled with values to complete the 32-bit register however because we had the instruction of mov al, 245 where al is 8-bit register and it handles the last two digits of the EAX register which is why we see F5 placed at the end which is shown F which is 15 in decimal 15 \* 16 = 240 + 5 = 245**

**A screenshot of a computer

Description automatically generated**

Line number: **9**

Instruction: **mov bl, 41**

Register value: **EBX = 00D6D029**

Explanation: **The EBX register is filled with values to complete the 32-bit general purpose register though because we focus on the last two digits of the EBX register, the mov bl, 41 instruction would output 41 in hexadecimal format which is why 29 is placed as the last two digits of the EBX register, 2 \* 16 = 32 + 9 = 41**

**A screenshot of a computer

Description automatically generated**

Line number: **10**

Instruction: **mov cl, 11**

Register value: **ECX = 00A1100B**

Explanation: **The ECX register is filled with values to complete the 32-bit general purpose register though the instruction mov cl, 11 would move the value 11 in hexadecimal which is why 0B is placed as the last two digits of the ECX register, 0 \* 16 = 0 + B (11) = B (11)**

**A screenshot of a computer

Description automatically generated**

Line number: **11**

Instruction: **mov dl, 215**

Register value: **EDX = 00A110D7**

Explanation: **The EDX register is filled with values to complete the 32-bit general purpose register though the instruction mov dl, 215 would move the value 215 in hexadecimal which is why D7 is placed as the last two digits of the EDX register, D (13) \* 16 = 208 + 7 = 215**

**A screenshot of a computer

Description automatically generated**

Line number: **12**

Instruction: **sub al, dl**

Register value: **EAX = 00B7FA1E**

Explanation: **The EAX register is filled with values to complete the 32-bit general purpose register however the instruction sub al, dl would subtract dl 8-bit register value which is D7 (215) from al 8-bit register value which is F5 (245), F5 (245) – D7 (215) = 1E which is (1\*16) = 16 + E (14), 1E = 30 and this is because the 5 borrows 1 from F becoming E and 5 becomes 16 + 5 = 21 now the equation is E 21 – D7 = 21 – 7 = 14 which is E and E – D = 14 – 13 = 1 therefore this is how we get the result 1E for the ending two digits of EAX register**

**A screenshot of a computer

Description automatically generated**

Line number: **13**

Instruction: **add al, cl**

Register value: **EAX = 00B7FA29**

Explanation: **The EAX register is filled with values to complete the 32-bit general purpose register however the instruction add al, cl would add cl 8-bit register value which is 0B (11) to the al 8-bit register value which had been updated from previous instruction to become 1E (30), 0B (11) + 1E (30) = when you add B (11) + E (14) = 25 (19) therefore it has to be divided by 16 because it takes up two bits and not one, 25 / 16 divides 1 time and the remainder is 9, place the one in the ones answer place and carry the 1 on top of 0 + 1 = 1 + the carry 1 = 2, the result is the last two digits of the EAX register being 29**

**A screenshot of a computer

Description automatically generated**

Line number: **14**

Instruction: **sub al, bl**

Register value: **EAX = 00B7FA00**

Explanation: **The EAX register has values to fill the 32-bit register towards the beginning however the last two digits become 0 because the instruction sub al, bl would subtract bl 8-bit register value which is 29 in hexadecimal and (41) in decimal from al 8-bit register value which is 29 in hexadecimal and (41) in decimal, the result would be 29 (41) – 29 (41) = 00 which means the last two digits of EAX and EBX were equal and when subtracted caused al to become 00**

**A computer screen shot of a black screen

Description automatically generated**