CSE340 Mid Solution

Q1 (Set A and B)

Briefly **list** the factors that impact the CPU performance and how they affect. **[2]**

Answer:

Performance depends on
Algorithm: affects IC, possibly CPI
Programming language: affects IC, CPI
Compiler: affects IC, CPI
Instruction set architecture: affects IC, CPI, Tc

Q2 (Set A and B)

The CPI of a program in an old processor is 6.0. The processor clock speed is 2 GHz. A new processor increases the clock speed to 4 GHz. However, the CPU time of the program in the new processor is worse than it was in the old processor. If the program requires 1000 instructions in the old processor and 1500 instructions in the new processor, then **calculate** the minimum value of CPI in the new processor. **[3]**

Answer:

CPU Time = (CPI * instruction count) / clock speed
Given that CPU time in the old processor is better than that in the new processor
CPU Time (new processor) / CPU Time (old processor) > 1
That is, $(CPI_{new} * 1500/4)/ (CPI_{old} * 1000/2) > 1$
That is, $(CPI_{new} * 1500 * 2) / (CPI_{old} * 1000 * 4) > 1$
That is, $(3000 * CPInew) / (4000 * CPI_{old}) > 1$
That is, $CPI_{new} > CPI_{old} * 4000/3000 = 6 * 4000 / 3000 = 8$
Hence, the answer is, the CPI has to be greater than 8 in the new processor.

Q3 (Set A and B)

Consider a system where you have data and instructions on the same memory. Now if you want to design a system where you have separate physical memory for data and instruction then list the challenges that you have to address. **[2]**

Answer:
The challenges could be 1. hardware cost 2. Design complexity 3. Since data and instruction can be of different lengths, managing the software level complexity can be a big challenge.

Q4 (Set A and B)

Suppose in the BRACU architecture's instruction format, similar to MIPS architecture, the opcode is 10-bit, and the funct field is 10-bit. **Calculate** how many R-type and J-type instructions can be generated considering the bit lengths given. Also, if each memory address can hold 16-bits and the architecture can support 32-bit addresses, then **calculate** the capacity of the memory. **[3]**

Answer:

For both R and J type, number of possible instructions: 2^{10} [1.5 Mark]

Capacity of memory = 2^{32} * 16 bits = 8 GB [1.5 Mark]

Q5 (Set A)

Construct the MIPS assembly code of the following code sequence. Assume the base of the array is in $s0, i is in $s1, the array is a byte array and i is a 32-bit integer. **[5]**
p

```
for (i = 0; i < 10; i++)
{
  array[i] = array[i + 1]
}
```

Answer:

```
ADD $s1, $zero, $zero
ADDI $t1, $zero, 10
LOOP:
SLT $t0, $s1, $t1
BEQ $t0, $zero, EXIT
ADD $t2, $s0, $s1
ADDi $t3, $t2, 1
LB $t4, 0($t3)
SB $t4 0($t2)
ADDi $s1, $s1, 1
J LOOP
EXIT:
```

Q5 (Set B)

Construct the MIPS assembly code of the following code sequence. Assume the base of the array is in $s1, i is in $s2, the array is a byte array and i is a 32-bit integer. **[5]**

```
for (i = 0; i < 15; i++)
{
  array[i] = array[i + 1]
}
```

Answer:

```
ADD $s2, $zero, $zero
ADDI $t1, $zero, 15
LOOP:
SLT $t0, $s2, $t1
BEQ $t0, $zero, EXIT
ADD $t2, $s2, $s1
ADDi $t3, $t2, 1
LB $t4, 0($t3)
SB $t4 0($t2)
ADDi $s2, $s2, 1
J LOOP
EXIT:
```

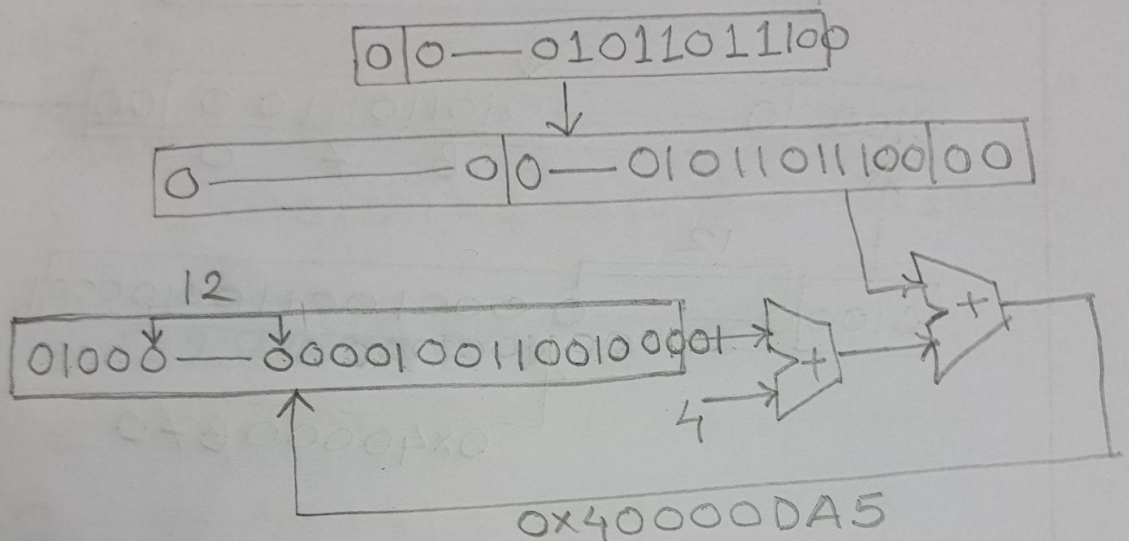
Q6 (Set A)

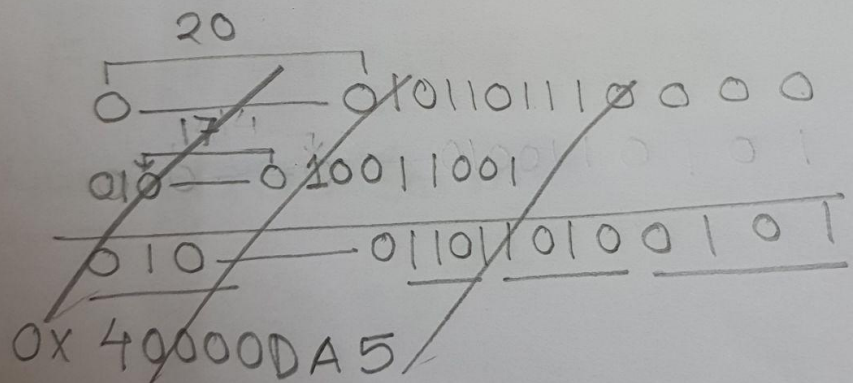
Consider that the PC has the value (in hex) 0x40001321. If the offset value(in decimal) is 732, then **calculate** the conditional and unconditional branch target address. (*Your answer should show all the steps associated with the calculation and the final result should be in hex format.)*

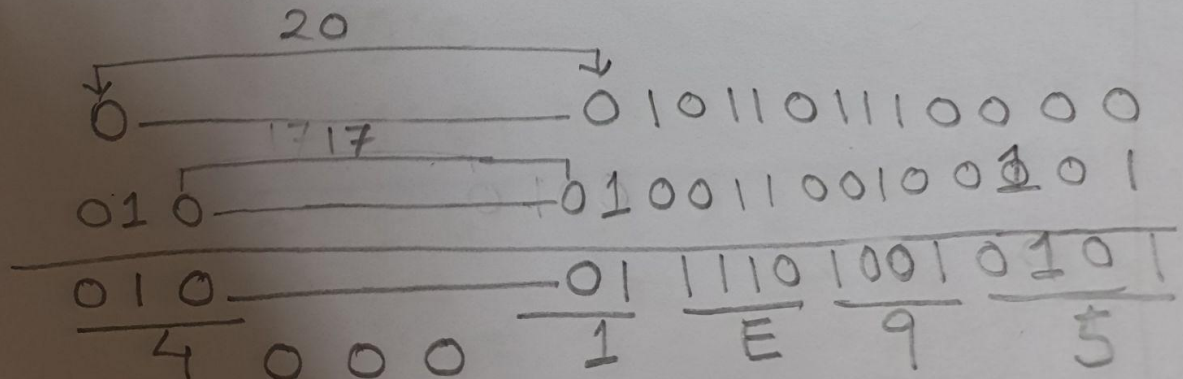
Answer:

06. PC: 0x40001321 ; Offset: 732

Conditional Branch

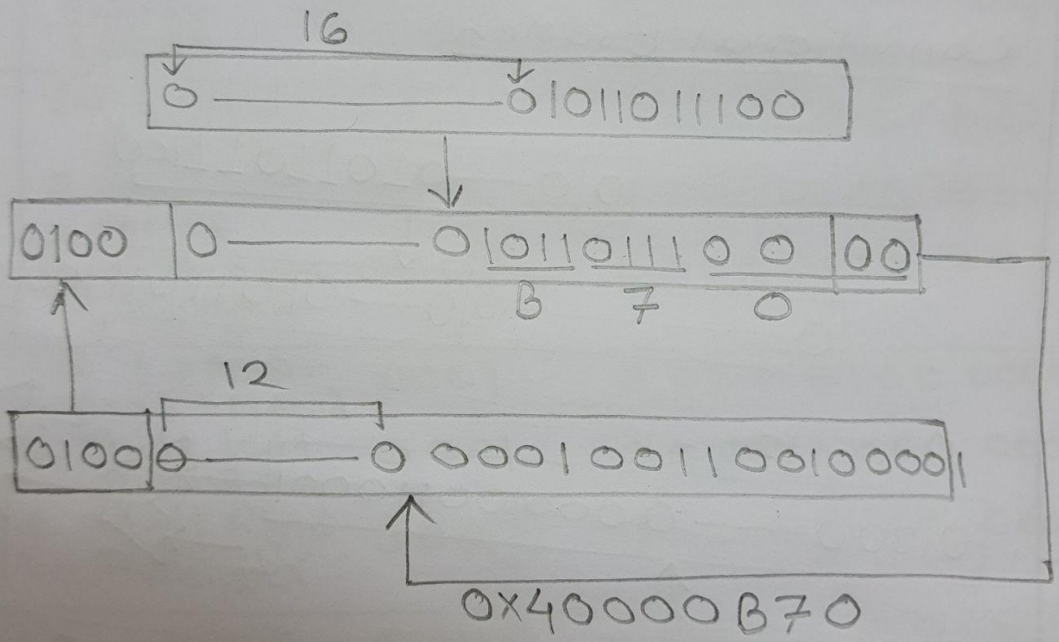






Unconditional Branch

PC: 0x40001321 ; offset: 732



Q6 (Set B)

Consider that the PC has the value(in hex) 0x40001222. If the offset value(in decimal) is 721, then **calculate** the conditional and unconditional branch target address. (*Your answer should show all the steps associated with the calculation and the final result should be in hex format.)* **[5 Marks]**

Answer:

Conditional branch address: 0x40001222 + 4 + (offset * 4) = 0x40001D6A

Unconditional branch address: 0x40000B44

Q7 (Set A and B)

Let's assume X= 4A2B0000 and Y=72BA0000 (Here X and Y are in IEEE-754 floating point format). Now, using IEEE-754 floating-point representation, **calculate** X+Y. **[5 Marks]**

Answer:

7. $X = 4A2B0000$

$= 0\overline{10010100}10101011 0 \text{——} 0$

So, $Q = 10010100 = 148$

So, $E = 148 - 127 = 21$

Normalized form

$1.01010111 \times 2^{21} = 0.\overbrace{0 \text{——} 0}^{80}10101011 \times 2^{10}$

$Y = 72BA0000$

$= 0\overline{11100101}01110100 \text{——} 0$

$Q = 11100101 = 229$

$E = 229 - 127 = 102$

Normalized form

1.011101×2^{102}

$X + Y$:

$0.0 \text{——} 010101011$

$1.0111010 \text{——} 0$

$\approx 1.011101 \times 2^{102}$

$= 1011101 = 2^{102} + 2^{100} + 2^{99} + 2^{98} + 2^{96}$

$= 7.3682 \times 10^{30}$