CMPT 334 **Spring 2021**

Assignment 4

Due: Wednesday April 21, 2021 – 8:00 AM

Submit a single pdf/word file in Moodle.

Provide short answers for the following question (100 points):

Q1. What is the function of an I/O module?

It is responsible for multiple tasks including:

- Control and timing.
- Processor communication.
- Device communication.
- Data buffering.
- Error detection.

Q2. List and define the three techniques for performing I/O

Programmed I/O: The processor issues an I/O command, on behalf of a process, to an I/O module; that process then busy-waits for the operation to be completed before proceeding.

Interrupt-driven I/O: The processor issues an I/O command on behalf of a process, continues to execute subsequent instructions, and is interrupted by the I/O module when the latter has completed its work. The subsequent instructions may be in the same process, if it is not necessary for that process to wait for the completion of the I/O. Otherwise, the process is suspended pending the interrupt and other work is performed.

Direct memory access (DMA): A DMA module controls the exchange of data between main memory and an I/O module. The processor sends a request for the transfer of a block of data to the DMA module and is interrupted only after the entire block has been transferred.

Q3. How does the processor determine which device issued an interrupt? (list the common techniques)

Four general categories of techniques are in common use: multiple interrupt lines; software poll; daisy chain (hardware poll, vectored); bus arbitration (vectored).

Q4. What is a machine instruction and what it consists of?

A machine instruction is a command given to the processor in order to perform a certain operation. It is the way in which a programmer communicates with the processor. The essential elements of a computer instruction are the opcode, which specifies the operation to be performed, the source and destination operand references, which specify the input and output locations for the operation, and a next instruction reference, which is usually implicit.

Q5. What is arithmetic shift and logical shift?

With a logical shift, the bits of a word are shifted left or right. On one end, the bit shifted out is lost. On the other end, a 0 is shifted in. The arithmetic shift operation treats the data as a signed integer and does not shift the sign bit. On a right arithmetic shift, the sign bit is replicated into the bit position to its right. On a left arithmetic shift, a logical left shift is performed on all bits but the sign bit, which is retained.

Q6. Why it is essential to have transfer control instructions?

1. In the practical use of computers, it is essential to be able to execute each instruction more than once and perhaps many thousands of times. It may require thousands or perhaps millions of instructions to implement an application. This would be unthinkable if each instruction had to be written out separately. If a table or a list of items is to be processed, a program loop is needed. One sequence of instructions is executed repeatedly to process all the data.
2. Virtually all programs involve some decision-making. We would like the computer to do one thing if one condition holds, and another thing if another condition holds.
3. To compose correctly a large or even medium-size computer program is an exceedingly difficult task. It helps if there are mechanisms for breaking the task up into smaller pieces that can be worked on one at a time.

Q7. What is the packed decimal representation of the following binary numbers. If you can't convert a particular binary number to packed decimal, describe why there will be an error.

- **1001 0011 0010**
- **0101 1000 0010**
- **0100 0110 1010**

932, 582, error (because packed decimal only represents digits from 0 to 9, 1010 is 10 in decimal)

Q8-10. Write a program to compute "X = (A + B * C) / D – E" using (Hint: use temporary register T):

A. One-address instructions given the following set of instructions:

LOAD M
STORE M
ADD M
SUB M
MUL M
DIV M

Answer:

```
LOAD   D
SUB    E
STORE  T
LOAD   B
MUL    C
ADD    A
DIV    T
STORE  X
```

B. Two-address instructions given the following set of instructions:

MOVE X, Y (X <- Y)
ADD X, Y (X <- X + Y)
SUB X, Y (X <- X - Y)
MUL X, Y (X <- X * Y)
DIV X, Y (X <- X / Y)

Answer:

```
MOVE   X, B    (X = B)
MUL    X, C    (X = B * C)
ADD    X, A    (X = A + B * C)
MOVE   T, D    (T = D)
SUB    T, E    (T = E)
DIV    X, T    (X = (A + B * C) / (D – E))
```

C. Three-address instructions given the following set of instructions:

MOVE X, Y (X <- Y)
ADD X, Y, Z (X <- Y + Z)
SUB X, Y, Z (X <- Y - Z)
MUL X, Y, Z (X <- Y * Z)
DIV X, Y, Z (X <- Y / Z)

Answer:

```
MUL    X, B, C   (X = B * C)
ADD    X, X, A   (X = A + B * C)
SUB    T, D, E   (T = D – E)
DIV    X, X, T   (X = A + (B * C) / (D – E))
```