## Problem 1:

Assume an architecture with a byte addressable memory. An instruction is stored at location 500 with its address field at location 502. The address field has a value of 300. The location 300 contains the value 140. Registers R1, R2, and R3 contain the values 85, 95, and 105 respectively. What is the effective address of the instruction operand for each of the following addressing modes? Show your work.

a. Direct Memory Addressing mode.
b. Indirect memory Addressing mode.
c. Immediate
d. PC-relative
e. Register Indirect if the location 502 contains the value 2 instead of 300.
f. Indexed assuming that R3 is the index register.

## Problem 2:

Assume an architecture with a byte addressable memory. An instruction is stored at location 300 with its address field at location 301. The address field has a value of 400. An accumulator register R1 contains the number 200. Evaluate effective address if the addressing mode of the instruction is:

a. Direct memory
b. Immediate
c. PC-Relative
d. Register indirect
e. Indexed with R1 as the index register.

## Problem 3:

A two-word instruction is stored in memory at an address designated by the symbol W. The address field of the instruction (stored at W+1) is designated by the symbol Y. The operand used during the execution of the instruction is stored at an address symbolized by Z. An index register contains the value X. State how Z is calculated from the other addresses if the addressing mode of the instruction is:

a. Direct memory
b. Immediate

c. PC-Relative

d. Indexed

## Problem 4:

A PC-Relative mode branch type of instruction is stored in memory at an address equivalent to decimal 750 (assume one cell per instruction). The branch is made to an address equivalent to decimal 500. What should be the value of the relative address field of the instruction (in decimal)?

## Problem 5:

A PC-Relative mode branch type of instruction is stored in memory at an address equivalent to decimal 1000 (assume two cells per instruction). The branch is made to an address equivalent to decimal 2000. What should be the value of the relative address field of the instruction (in decimal)?

## Problem 6:

Consider a hypothetical 24-bit microprocessor having 24-bit instructions composed of two fields: The first byte contains the Opcode and the remainder the operand address. What is the maximum directly addressable memory capacity (in bytes)?

## Problem 7:

A computer has a 32-bit instructions and 12-bit addresses. If the instructions on this computer are either of type one-address instructions or two-address. If there are 250 two-address instructions, how many one-address instructions can be formulated?

## Problem 8:

Consider a hypothetical 32-bit microprocessor having 32-bit instructions composed of two fields: the first byte contains the opcode and the remainder the immediate operand or an operand address.

a) What is the maximum directly addressable capacity (in bytes)?

b) Discuss the impact on the system speed if the microprocessor bus has:

   1. A 32 bit local address bus and a 16-bit local data bus.
   2. A 16 bit local address bus and a 16-bit local data bus.

c) How many bits are needed for the PC and the IR registers?

## Problem 9:

Discuss briefly the following MIPS addressing modes. State an example for each one of them.

a) Register direct addressing mode

b) PC-relative addressing mode

c) Base-addressing mode

d) Pseudo-direct addressing mode

## Problem 10:

The MIPS architecture supports some and doesn't support some of the addressing modes below. Synthesize these instructions by writing one or more MIPS lines that do the same functionality.

a) Load $s1, ($s2) // Register Indirect addressing mode.

a) Load $s1, @($s2) // Memory Indirect addressing mode, assuming that $S2 contains the first memory address

b) Load $s1, ($s2+$s3+200) // indexed with displacement addressing mode.

c) Load $s1, (0x1122AABB) // memory direct addressing mode.

## Problem 11:

Assume in the MIPS PC register has the value of 0x1122AABB, a j instruction has an address field of 0x003AABB. What is the address that the jump instruction would like to reach?

## Problem 12:

Given the following MIPS code, state the addressing mode in each of its lines:

```
loop: addi  $s1, $s2, 60
      add $s3, $s1, $t0
      lw $t4, 4($s3)
      j loop
```