

Team 2-1

Maximum Points : 10 / 10

Grading criteria for Pre-Milestone 1

1. Completeness of assembly language instructions (7 points)
 - a. Instructions to access memory (1 point)
 - Effective address (register indirect/direct/base-displacement)
Lv and sv.

 - b. Instructions to do basic arithmetic (add, sub, logical) (1 point)
 - Number of operands
 - immediates
 - size of immediate field
 - memory operands (address calculation)
yeah. As a note: mult and div are difficult to implement, things are easier without them, you may choose to try it if you wish, but I suggest against it.

 - c. Instructions to conditionally branch (2 points)
 - MIPS-like “condition-code less”
 - Use of condition-codes
 - Effective address (register indirect/direct/base-displacement)
 - If register indirect/base-displacement, is there any way to load an entire address into the register
 - Size of address
Ceq and cne.

 - d. Instructions to un-conditionally branch (1 point)
 - Effective address (register indirect/direct/base-displacement)
 - If register indirect/base-displacement, is there any way to load an entire address into the register
 - Size of address
Goal, go, and gor. Nice.

- e. Instructions to handle procedures (1 point)
 - instruction to transfer control
 - instruction to return to calling proceduregoal and gor.

- f. Other data movement instructions (1 point)
 - between registers
 - immediate value to register (size of immediate value)move, luc, and orc.

2. Program for Relatively prime value (3 points)

- a. Uses instructions from above-said list, not MIPS. (-3 if MIPS, don't continue.)
- b. Tests approximately 55 % of the above-listed instructions.
- c. Input values read from memory and output written to memory. (1 pt)

Yes.

- d. Procedure used to determine the gcd using Euclid's algorithm as specified in the high-level language. (1 pt)

It would appear so.

- e. Use of conditional instructions in main/procedure. (1 pt)

Yes.

- f. Any conventions followed regarding parameter passing and returning values,
- g. Approximate number of registers the program uses.