Show all work necessary to justify your answers!

1. If parameters are passed by value, what will the following program output with:

   - static scoping,
   - dynamic scoping

   for non-local variables.

```
program P;
  var x, y: integer;

  procedure Q;
  begin
    x := y * 4;
  end;

  procedure R;
    var x: integer;
  begin
    x := y + 3;
    y := 2 * y;
    Q;
    writeln(x);
  end;

  x := 2;
  y := 5;
  R;
  writeln(x, y);
end.
```

2. What will the following program output if parameters are passed with:

- call-by-value
- call-by-value-result
- call-by-reference
- call-by-name (macro expansion)

```
program S;
  var x, y, j: integer;

  procedure T (y, z :integer);
  begin
    z := z - 5;
    y := y + 5;
    x := x - y;
  end;
begin
  x := 3;
  y := 4;
  T(x, y);
  writeln (x, y);
end.
```

3. This question requires you to write pseudo-code for several MIPS assembly language in-
   structions that must be executed to run the following program. Your pseudo-code should be
   as detailed as the assembly instructions, but we are not concerned about syntax. For example,
   the instruction "sw $fp, 0($sp)" can be written as "store the frame pointer at 0 past the stack
   pointer."

```
program P1;
   var A,B: integer;

   procedure F(N, Total: integer);

      procedure Print(Value: integer);
      begin
         writeln(Value, Total);
      end;

   begin
      if (N >= 1) then
      begin
         Total := Total * N;
         N := N - 1;
         F(N, Total)
      end
      else
         Print(Total)
   end;

begin
   A := 5;
   B := 1;
   F(A, B);
end.
```
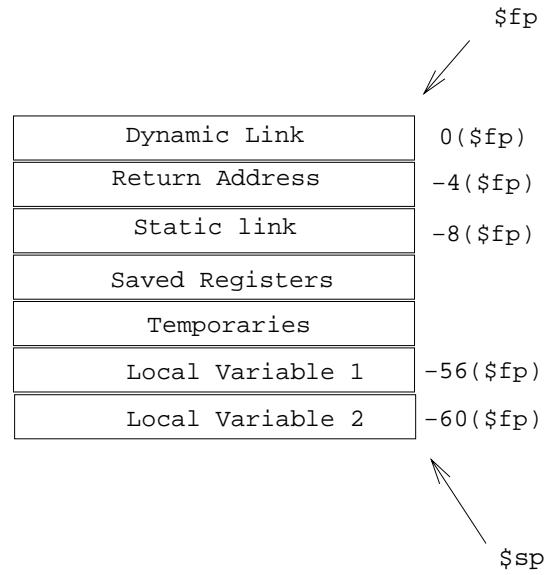
You should assume the register and parameter passing conventions discussed in class. The
layout of a single stack frame (activation record) is shown below.

```
                                                    $fp
                                                     ↙

        ┌─────────────────────────┐
        │      Dynamic Link       │      0($fp)
        ├─────────────────────────┤
        │     Return Address      │      -4($fp)
        ├─────────────────────────┤
        │       Static link       │      -8($fp)
        ├─────────────────────────┤
        │     Saved Registers     │
        ├─────────────────────────┤
        │       Temporaries       │
        ├─────────────────────────┤
        │    Local Variable 1     │      -56($fp)
        ├─────────────────────────┤
        │    Local Variable 2     │      -60($fp)
        └─────────────────────────┘

                                            ↖
                                              $sp
```

(a) Provide the pseudo-code that is necessary to set up the runtime stack when procedure
    F calls itself recursively, for both the caller and the callee.

(b) What does the program print?