# CS352 Compilers: Principle and Practice
# Spring 2005

## PSO on Monday, April 25, 2005

**Mummoorthy M**

*Some are from Bernice's notes

---

**Agenda:**

1. Course Evaluations
2. Project 6
3. Closing Remarks

---

### 1. Evaluations for this PSO

Need a volunteer

### 2. Project 6:

*As Prof. Hosking puts it, the final piece of the puzzle

*Conversion of IR Tree to Actual PowerPC instructions

For example,

the Tree.BINOP node has

public Exp left, right;

If e is an Tree.ADD then

You can output 'add' instruction

*These transformations can't be straightforward due to various reasons. Consider the following:

i=i+1

you can't blindly output 'add' instruction. Here you need 'addi' which is a constant addition i.e. 'add immediate'

Similarly you can do code transformations for

1. MOVE
2. JUMP ; CJUMP
3. BINOP; BINOP.ADD, BINOP.SUB, BINOP.MUL, BINOP.DIV
   BINOP.AND, BINOP.OR, BINOP.XOR
4. MEM
5. CALL (*)

Note: '#' is for comments. That means the instructions that have '#' as the starting letter are comments.

**Let us look at some of the useful examples:**

**1. Add Immediate**

```
P5:   i=i+1;
      addi r3,r3,1
```

```
C9:   i=i+1;
      add r3,r3,r3
```

```
P5:   i=1+2

      li r3,1
      addi r3,r3,2
```

C9's output is obviously wrong. Can you spot why it is wrong?
Note that P5 looks at the arguments and decides to output 'addi' instead of 'add'.

Similarly BINOP instructions can be corrected using this simple observation.
How about i=1+i?

**2. MUL**

```
P5:
        i=i*i;
        i=i*16;
        i=i*18;


         mullw r3,r3,r3
         slwi r3,r3,4
         mulli r3,r3,18
```

Why for i*16 you are seeing some strange instructions? Can you spot any pattern for doing this?

If you have studied Computer Architecture, you must know this optimization. Similarly try for DIV operation

3. MOV(dst, src)

Current implementation does not check if offset is CONST.

4. JUMP and CJUMP

5. CALL

-> Arguments to corresponding registers

-> JUMP instruction or ‘bctrl’

**3. Closing Remarks**

*Best Wishes for all of your future endeavors

*Learning is crucial. Hope you continue learning new things with a vigor of a class environment, wherever you go.

*Try to use the ideas we learned here.

- Natural Language Processing
- Software Design
- Testing
- Attending PSOs ☺, asking tough questions to TA's
- & so many…

# How about writing a 'C' Compiler in summer?

☺