Milestone Presenation

Team 4 : 김도형, 정재환, Alena Kazakova

4190.570 Advanced Compiler Construction

Implementation Progress

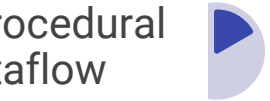
Modified ir.cpp to generate CFG from TAC (one instruction per block)

Done

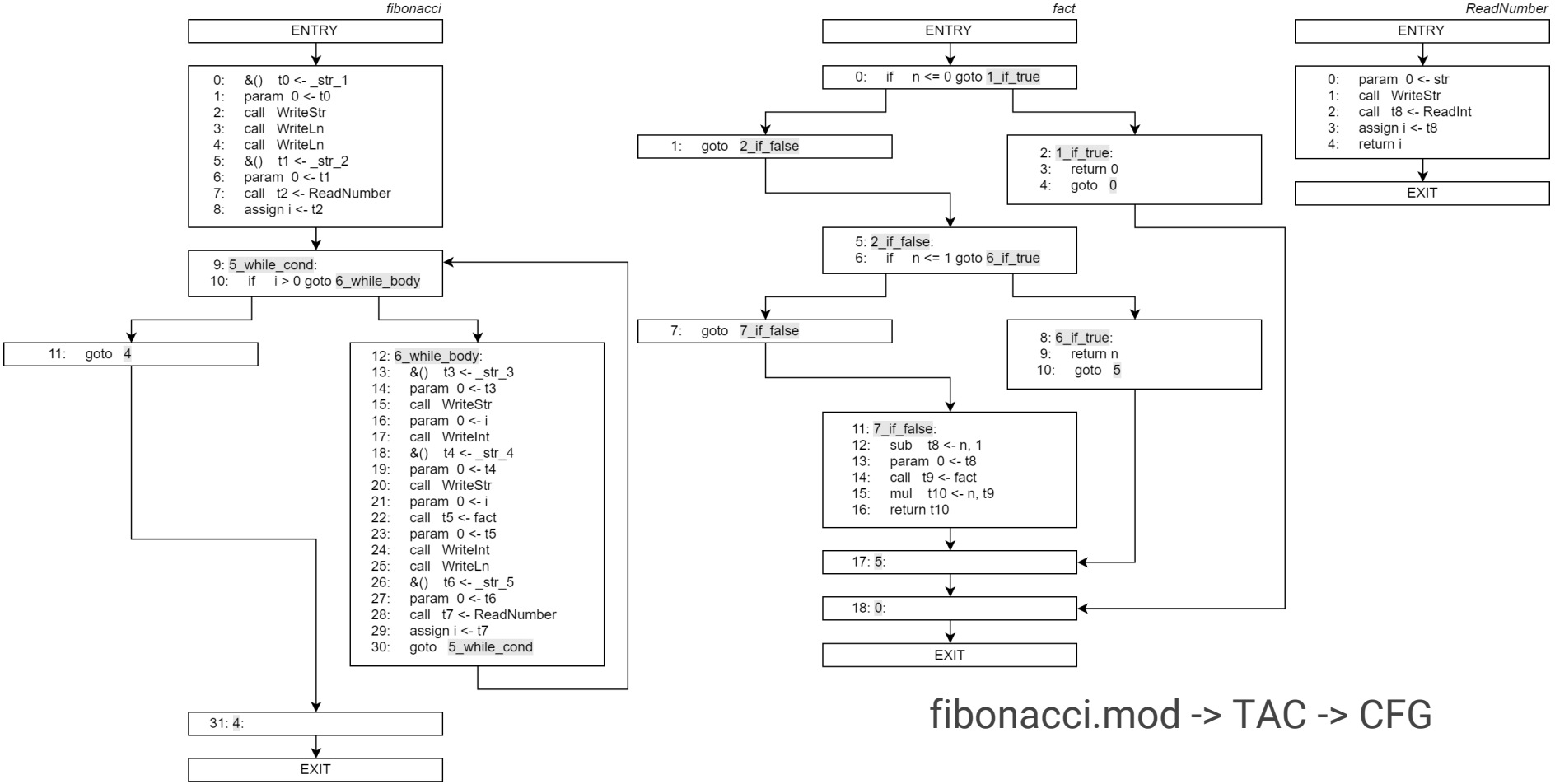
Dataflow analysis: constant propagation and dead code elimination

Work In Progress

Generate interprocedural CFG and run dataflow analysis again



Ideation



fibonacci.mod -> TAC -> CFG

Difficulties

- How does snuplc work?
- How is TAC generated?

Solution

- Breakpoint Debugging
- Printing TAC using public functions

```
void PrintInstr(CTacInstr *instr) {
    unsigned int id = instr->GetId();
    EOperation op = instr->GetOperation();
    CTac *dst = instr->GetDest();
    CTacAddr *src1 = instr->GetSrc(1);
    CTacAddr *src2 = instr->GetSrc(2);
    CTacLabel *lbl = dynamic_cast<CTacLabel*>(instr);
    bool relop = IsRelOp(op);
    cout << right << dec << setw(3) << id << ": ";
    if (op == opLabel) cout << lbl->GetLabel() << ": ";
    else {
        cout << "    " << left << setw(6);
        if (relop) cout << "if";
        else cout << op;
        cout << " ";
    }
    CTacAddr *adr = dynamic_cast<CTacAddr*>(dst);
    if (adr != NULL) cout << dst << " <- ";
    if (src1 != NULL) cout << src1;
    if (src2 != NULL) {
        if (relop) cout << " " << op;
        else cout << ",";
        cout << " " << src2;
    }
    CTacInstr *target = dynamic_cast<CTacInstr*>(dst);
    if (target != NULL) {
        if (relop) cout << " goto ";
        lbl = dynamic_cast<CTacLabel*>(target);
        if (lbl != NULL) cout << lbl->GetLabel();
        else cout << target->GetId();
    }
}
```

Project Plan

	October				November				December	
	Week 1	Week 2	Week 3	Week 4	Week 1	Week 2	Week 3	Week 4	Week 1	Week 2
Dataflow Construction										
Optimization Implementation										
Extension Implementation										
Debugging & Modifications										
Implementation Evaluation										
Final Report & Presentation										