

The following are my code snippets and comments for each function I implemented. I just list a few functions that make me struggled while debugging. However, there is nothing complex in this assignment, so I just post the code of LICM for it is more challenging.

- **LICM.cpp/isSafeToHoistInstr()**

This function checks whether it is safe to hoist the given instruction.

```
bool LICM::isSafeToHoistInstr(llvm::Instruction *Inst)
{
    // cond1 = has loop invariant operands
    // cond2 = safe to speculatively execute
    // cond3 = one of the following instruction classes:
    //          BinaryOperator, CastInst, SelectInst, GetElementPtrInst, and CmpInst
    bool cond1 = false, cond2 = false, cond3 = false;
    if (mCurrLoop->hasLoopInvariantOperands(Inst)) cond1 = true;
    if (isSafeToSpeculativelyExecute(Inst)) cond2 = true;

    if (isa<BinaryOperator>(Inst)) cond3 = true;
    else if (isa<CastInst>(Inst)) cond3 = true;
    else if (isa<SelectInst>(Inst)) cond3 = true;
    else if (isa<GetElementPtrInst>(Inst)) cond3 = true;
    else if (isa<CmpInst>(Inst)) cond3 = true;

    if (cond1 && cond2 && cond3) return true;
    else return false;
}
```

- **LICM.cpp/hoistInstr()**

The function hoists the given instruction.

```
void LICM::hoistInstr(llvm::Instruction *Inst)
{
    // hoists the instruction to the preheader block
    Inst->moveBefore(mCurrLoop->getLoopPreheader()->getTerminator());
    mChanged = true;
}
```

- **LICM.cpp/hoistPreOrder()**

The function hoists instructions from the given DTnode to its children.

```
void LICM::hoistPreOrder(llvm::DomTreeNode *DTNode)
{
    BasicBlock *bb = DTNode->getBlock();

    // if BB is in the current loop...
    if (mCurrLoop == mLoopInfo->getLoopFor(bb))
    {
        BasicBlock::iterator it = bb->begin(), currentInstr;
        while (it != bb->end())
        {
            currentInstr = it;
            it++;
            if (isSafeToHoistInstr(currentInstr))
                hoistInstr(currentInstr);
        }
    }
    std::vector<DomTreeNode*> DTChildren = DTNode->getChildren();
    for (auto& child : DTChildren)
        hoistPreOrder(child);
}
```