Project 4: Global Common Subexpression Elimination

Due Date: 04/04/2017

Programs should be submitted through CSNET by the announced time. Submissions should be in the form of a single file named `project4_UIN.tar.gz` file (where UIN is your UIN number) containing the relevant source code and a README file explaining your approach (see Creating .tar.gz File below on how to create and submit this file.)

Introduction

This project will implement another data flow analysis algorithm using the MiniPolaris compiler. However, this time your input already is in SSA form. This will make it easier to implement since the def-use chains are known beforehand.

The algorithm you will implement in Project 4 is *global common subexpression elimination (GCSE)*. To simplify design, the project is broken into three parts :

1. Calculate available expressions. (35%)
2. GCSE. (45 %)
3. Copy propagation. (20%)

Input and Output

Input

The input to your program will be a FORTRAN 77 program. This will be specified on the command line when running your copy of MiniPolaris, e.g.

```
mini_polaris -f sw test.f > test.list
```

In this example, `test.f` is a FORTRAN program which you want to compile.

We save the output to the file `test.list` which may be examined for correctness. The file `sw` contains your own copy of the switchfile in your work directory. (see Switches.)

Output

The output that MiniPolaris produces goes to `stdout` (error messages to `stderr`) and is a listing containing information produced by the compiler during each pass of the compiler. Converting this list file to an actual FORTRAN file is done as follows :

```
list2src < test.list > test.f
```

The output FORTRAN program produced by project 4 should contain all the statements of the original, with the exception of deadcoded statements. The driver program will provide the option of printing out the current subprogram just after you have completed the pass, either in normal FORTRAN or a detailed listing of the abstract syntax tree. (see Switches.)

Program Design

Switches

All the routines for global common subexpression elimination should be controlled by switches (such as the file `sw` used in the above example.) For this pass you will use the following switches :