Assignment 4

1. Conditional Inter-Procedural Precise Copy Analysis

Modify the Inter-Procedural Precise Copy Analysis to take the values of conditions into account.

- Specify a support function `evalExp` to evaluate boolean expressions and use this support function to determine whether a condition is evaluated to true, false, or both results are possible.
- Specify two transfer functions associated with boolean expressions, one to be used in the case the true branch is followed and one to be used in the case the false branch is followed.

Relevant statements are

- IfStmt(ExprStatement(cond)), true_edge:
- IfStmt(ExprStatement(cond)), false_edge:
- WhileStmt(ExprStatement(cond)), true_edge:
- WhileStmt(ExprStatement(cond)), false_edge:
- LogicalIf(cond), true_edge:
- LogicalIf(cond), false_edge:

2. Analysis Information Size

a) Let us consider sequences of SL2 assignments of the form $x_2 = x_1; x_3 = x_2; \ldots; x_n := x_{n-1};$. Give a formula for the analysis information size of the last assignment's exit-information of your analysis.

b) Let an SL2 program P consist of functions of the form

$$\texttt{int } f_i(x_i)\{\texttt{int } x_{i+1} = x_i; \texttt{return } f_{i+1}(x_{i+1});\}$$

with $1 \le i \le n$, the function

$$\texttt{int } f_{n+1}(x_{n+1})\{\texttt{return } x_{n+1};\}$$

and the main function. The main function consists of an assignment $x_0 = 1$, the assignment $z = f_1(x_0)$, and the necessary variable declarations. Give a formula for the size of the exit-information of the assignment $x_{n+1} = x_n$ in function f_n of your analysis.

Note that call and return nodes provide information on the function name and actual parameters

- FunctionCall(fname,actualParametersList),call_edge:
- FunctionCall(fname,actualParametersList),local_edge:
- FunctionReturn(fname,actualParametersList),normal_edge:

where fname is of type `str` and actualParametersList of type `*VariableSymbolNT`, meaning that there can be an arbitrary number of VariableSymbols, i.e. parameters. For an example on how to process such lists see the support function `kill_vars` in the sl2rd example analysis.

3. Precision of the Analysis

Investigate the precision of your analysis with respect to

a) Conditionals in if-statements and while-statements. Write two example programs where the analysis results are more precise when taking conditionals into account than without doing so. Explain the difference in precision by comparing the analysis results for one particular node of your choice in the respective ICFG.

b) Non-recursive programs and call strings. Write one example program for which the analysis results are different for call string length 0,1, and 2. Explain the difference in precision by comparing the three different analysis results for one particular node of your choice in the ICFG.

c) Recursive programs and call strings. Write one recursive example program and discuss for which call string length the precision of your analysis does not improve anymore for program locations in the function `main`. Ensure that that the precision is different with call string length 0 and 1 in function main.

Note that changing the underlying lattice (carrier) of the Inter-Procedural Precise Copy Analysis is encouraged but not mandatory. Ensure that your analysis terminates for any given SL2 program. Refer to the help message of the generated analyzer on how you can run the analyzer with call strings of different length.

4. Hand in

- Send your answers, example programs, and PAG specification per e-mail to `markus@complang.tuwien.ac.at`
- The e-mail must have as subject "OPTUB:Assignment 4, <LastName>" where <LastName> is replaced with your last name.
- Deadline: 2pm December 5, 2007.