Leann Estrada

GATech CS6340

Fall 2016

Assignment 7

1. There are two bugs represented in this report. This is because of two reasons. Firstly, there are two top-level predicates. Secondly, upon reviewing each of the affinity lists for the two top-level predicates, most of the effective scores of the predicates are zero, indicating they refer to the same bugs.

2.

asgn7$ . ./src/jpegtran -outfile testout.jpg -flip vertical testimg.jpg

Segmentation fault: 11

asgn7$ ./src/jpegtran -outfile testout.jpg -transverse testimg.jpg

Bogus virtual array access

3.

The first top-level predicate associated with a bug is “info->transform == 2 is TRUE”, found in function jtransform\_request\_workspace (transupp.c, line 585). When that predicate is true, the variable info->workspace\_coef\_arrays is set to NULL. Later in the execution, in the function jtransform\_execute\_transformation (transupp.c), the info->workspace\_coef\_arrays variable is used to initialize the variable dst\_coef\_arrays. The variable dst\_coef\_arrays is then passed to do\_flip\_v (transupp.c), the transform function that is called when the predicate is true, or in other words when the user passed the “-flip vertical” option. Within do\_flip\_v, elements of dst\_coef\_arrays are accessed (line 142). However, since dst\_coef\_arrays is NULL, this throws a memory access error. It is important to note that the no flip and flip horizontal options do not require the use of dst\_coef\_arrays, and this is why those options do not also throw an error even though they also pass through the if statement where the top-level predicate is found. To fix this bug, info->workspace\_coef\_arrays must be properly initialized when the user passes the “-flip vertical” option.

The second top-level predicate associated with a bug is “info->transform == 4 is TRUE”, found in function jtransform\_request\_workspace (transupp.c, line 592). When that predicate is true, the variable info->workspace\_coef\_arrays is set “having same dimensions as source image”. Later in the execution, in the function jtransform\_execute\_transformation (transupp.c), the info->workspace\_coef\_arrays variable is used to initialize the variable dst\_coef\_arrays. The variable dst\_coef\_arrays is then passed to do\_transverse (transupp.c), the transform function that is called when the predicate is true, or in other words when the user passed the “-transverse” option. Within do\_transverse, the variable dst\_buffer is allocated using the function access\_virt\_barray (jmemmgr.c), passing in dst\_coef\_arrays of the current component as ptr. Within access\_virt\_barray, the “Bogus virtual array access” error is thrown (line 853) because the first check, or if statement, fails, namely the predicate “end\_row > ptr->rows\_in\_array”. ptr->rows\_in\_array is incorrectly set because jtransform\_request\_workspace set the destination dimensions to be the same as those of the source image. However, the transverse operation requires a transposition operation, which by definition inverts the dimensions. Thus, the row that is attempting to be accessed in the virtual array does not exist, resulting in the error. To fix this bug, info->workspace\_coef\_arrays must be properly initialized with the maximum dimension of the source image as both dimensions when the user passes the “-transverse” option.