CS 406 Software Engineering

Fall 1997, October 20 Tool-Based Assignment #3

Due: In class, November 14

1 The problem

You are given a program P (sort.c) (located at : /home/champion/v/singha/sort.c).

Your group is required to develop four test sets for this program with the help of ATAC. These test sets, named BLOCK, DEC, PUSE, and CUSE, are block, decision, p-use and c-use adequate, respectively. Each test set consists of zero or more test cases. Each test case consists of a value of N, the number of elements in array A, and the elements of A itself. Try to keep each of your test sets as small in size as possible. If, for example, BLOCK is your block adequate test set consisting of k test cases, and then it is not possible to find another block adequate test set containing less than k test cases, then BLOCK is the smallest possible test set that satisfies the block coverage criterion. Smaller test tests will get higher grade!

While developing your test sets, begin by obtaining BLOCK. Your goal here is to develop the smallest (minimal) test set that provides 100% block coverage *without causing P to fail*. Thus, for example, if you design a test case that increases the block coverage but causes P to fail, then you need to ignore this test case and develop a new one. In case you are finding it *impossible* to obtain 100% block coverage without P failing on at least one test case then go ahead and include such a test case in BLOCK. If you are wrong in your judgement then you loose points! If you find that 100% coverage cannot be obtained due to infeasible paths, then turnin test sets that achieve the highest possible coverage.

After having determined BLOCK, add test cases, as many as required, to BLOCK to obtain DEC such that DEC leads to 100% decision coverage. Then add test cases to DEC to obtain PUSE that gives 100% p-use coverage. Finally add test cases to PUSE to obtain CUSE that gives 100% c-use coverage. Thus your test sets *must* satisfy the relation: BLOCK $\subseteq$ DEC $\subseteq$ PUSE $\subseteq$ CUSE.

While developing the test sets keep in mind the goal that these test sets be as small as possible and should *not* cause P to fail. However, if you find it *impossible* to do so then go ahead and derive the test sets anyway.

2 What to Submit ?

Your group needs to turnin the four test sets and the coverages obtained. The TA will execute the test cases to obtain the coverages and check if the coverages are 100% and the program does not fail. If the program fails on any test case, then you loose points unless you have *correctly* concluded that 100% coverage is impossible without P failing on at least one test case.