**Homework:**

(You may spend 10~20 hours for HW)

1. (90 pts) To write down kcov-branch-identify using Clang based on the provided template C++ file. kcov-branch-identify receives a file name of a single C file and prints the list of the branches at source code level as they are and the total number of branches of the C file. You have to submit your kcov-branch-identify code both in hardcopy and softcopy.

See the following output for the attached example-kcov.c:

$ ./PrintBranches example-kcov.c

function: f2

 If ID: 0 Line: 4 Col: 2 Filename: ./example-kcov.h

function: f1

 If ID: 1 Line: 19 Col: 2 Filename: example-kcov.c

function: main

 If ID: 2 Line: 30 Col: 2 Filename: example-kcov.c

 If ID: 3 Line: 32 Col: 9 Filename: example-kcov.c

 For ID: 4 Line: 39 Col: 2 Filename: example-kcov.c

 While ID: 5 Line: 44 Col: 2 Filename: example-kcov.c

 Do ID: 6 Line: 49 Col: 2 Filename: example-kcov.c

 Case ID: 7 Line: 51 Col: 4 Filename: example-kcov.c

 Case ID: 8 Line: 54 Col: 4 Filename: example-kcov.c

 ?: ID: 9 Line: 55 Col: 9 Filename: example-kcov.c

 Default ID: 10 Line: 58 Col: 4 Filename: example-kcov.c

 If ID: 11 Line: 63 Col: 2 Filename: example-kcov.c

 ?: ID: 12 Line: 63 Col: 7 Filename: example-kcov.c

 ImpDef. ID: 13 Line: 67 Col: 2 Filename: example-kcov.c

 Case ID: 14 Line: 68 Col: 3 Filename: example-kcov.c

 Case ID: 15 Line: 71 Col: 3 Filename: example-kcov.c

 Do ID: 16 Line: 76 Col: 2 Filename: example-kcov.c

 If ID: 17 Line: 76 Col: 2 Filename: example-kcov.c

Total number of branches: 30

Note 1. We count each case as one branch (i.e., considering switch(){…} has multiple outgoing edges). Also, we count (implicit) default statement as one branch regardless of whether default exists or not. A line and a column of an implicit default is those of corresponding switch().

Note 2. If a macro contains a branch, location information (a line, a column, and a filename) of the branch is those where the macro is used at.

1. (10 pts) To print out the branches in the attached grep file (i.e., grep.c) by using your kcov-branch-identify. Submit the entire output only in softcopy to save papers. Just report total # of branches in your HW hardcopy.

Note 1. If your program fails to find header files of a target program (grep.c), you have to modify include\_paths (line 118) in the initialization part of the kcov-branch-identify.cpp template file.

Note 2. You can ignore various Clang warnings.

Note 3. Your program should print out functions which have no branches.

Note 4. The total # of branches of the grep C file: > 3000

1. (90 pts) To write down kcov using Clang. You have to submit your kcov code both in hardcopy and softcopy.
	1. kcov receives a file name of a *preprocessed* single C file <f>.c and generates the instrumented version <f>-cov.c to measure branch coverage of <f>.c through testing.
		1. A preprocessed C file can be obtained by gcc –undef –E <filename>.c
	2. When <f>-cov.c is compiled and executed 1st time, <f>-cov.c generates a coverage measurement file <f>-cov-measure.txt. After then, <f>-cov.c updates <f>-cov-measure.txt through testing <f>-cov.c. The format of <f>-cov-measure.txt is as follows

Line# |# of execution |# of execution | conditional

 |of then branch |of else branch | expression

1453 0 0 errnum

1474 0 7 size && !result

1484 3 0 ptr

1488 0 0 size && !result

…

6950 0 0 (end = memchr(beg + len, '\n', (buf + size) - (beg + len))) != 0

6955 0 0 beg > buf && beg[-1] != '\n'

Covered: 581 / Total: 3101 = 18.735892%

Note1. If one line has multiple branches (i.e., nested if statements), you can print out these branches in separate lines with the same line id

Note2. The # of execution of else branch of case should be always 0 (i.e., meaningless)

Note3. For a switch statement, your program should print out case and (implicit) default statements. A conditional expression of case statement is a corresponding case value and that of default is ``default’’

1. (10 pts) To print out the coverage measurement file of the *preprocessed* grep C code with the following test cases where grep.c is the grep C file (not preprocessed C file)

Submit the output only in softcopy to save papers. Just report the last line of the measurement file (i.e., # of covered branch, # of total branch, and achieved branch coverage percentage) in your HW hardcopy.

./grep –n "if" grep.c
./grep –E "[0-9][0-9]+" grep.c
./grep -E “[[:digit:]][[:alpha:]]” grep.c