COM S 413x/513x Project 1: Understanding bugs **1**
Learning Objectives 1
Description 1
Deliverables 2
Extra Credit (15 pt) 2

COM S 413x/513x Project 1: Understanding bugs

Learning Objectives

1. Get further familiar with the terminologies related to bugs
2. Understand the challenge of bug understanding and the need of automatic tool support
3. Improve problem solving skills on code inspection and bug diagnosis
4. Work with real-world software and bugs

Description

In this homework, you are going to exercise your code inspection skills to analyze and understand bugs. As a first step, you will download the dbg benchmark located here: https://dbgbench.github.io/ This benchmark documented a set of real-world bugs from unix utilities *find* and *grep*. Here are some guidance on how you can explore this website effectively:

1. You can start under “use and reproduce”, and follow “Step-by-Step instructions”. The website also lists a set of tutorial materials (Click “tutorial materials” under “setup and infrastructure”) that can help you further understand the two programs.
2. Click “Data for find” and “Data for grep” on the website, you will find detailed explanations of the bugs, including the root causes, symptoms and the tests that can trigger the bugs. There are also a set of patches for each bug developed by different participants of the research as well as the patch from the developer’s original fix. You can find further information about the benchmark following the paper “Where is the bug and how is it fixed? an experiment with practitioners” https://dl.acm.org/citation.cfm?id=3106255

In this assignment, you are going to study and reproduce one of these bugs of your selection. Here are the list of steps to follow:

1. (5 pt) Study “simplified bug report” listed in the bug description, and create a data entry consisting of “program, bugid, bug type, bug location, fault location, fault condition,

failure location, patch location, test input that triggers the bug" and save it in a file called bug.txt. Note that the terminologies of bug and fault in the dbg benchmark may be different from the ones from the ones taught in our class. Please use the terminologies taught in our class.

2. (5 pt) Identify the cause of the bug and create a fault signature. For this question, you will need to submit two .c/.cpp files, the original buggy file, and the files that contain the fault signature. Also please write your answer for the question: "can the input that triggers the original bug also crash the fault signature?" in your readme.txt
3. (3 pt) Compute the the distances between bug and fault, fault and failure, failure and patch. You can measure them using lines of code. List the three distances in a file called distance.txt and separate the three distances using ","s.
4. (2 pt) What are your thoughts/findings from analyzing this bug?

If you are interested to evaluate your ability of bug diagnosis, you can also fill the questionnaire located https://dbgbench.github.io/questionnaire.pdf Then you can compare your answers with the survey results the researchers had generated over a set of experienced participates in the paper.

Deliverables

Please zip the following files and submit the zipped file to canvas under the "project 1" column.
bug.txt for Q1
distance.txt for Q3
readme.txt for Q2 and Q4
two .c/.cpp files

The homework is due Feb 12, Tue 6:00pm. You can continuously submit your homework without a penalty after the deadline. But once I started grading the homework, I am no longer accepting late homework or modifications. This project can be challenging if it is your first time for handling real-world bugs. So start early!

Extra Credit (15 pt)

If you are interested in further challenges, you can do the above assignment for an interesting real-world bug that is not documented in the dbg benchmark. Your findings can be submitted to as an expansion of dbg benchmarks.