

SOFT410911: Software Quality Assurance 2022-Fall

Homework-1 Solutions

软件 2101 杨豪 学号: 2206213297

2022 年 9 月 12 日

Honor Code: I promise that I finished the homework solutions on my own without copying other people's work.

BUG collection and reflect

BUG1. Compiler Output

1.1 bug itself

Yesterday I found a weird phenomenon that Latex worship, an extension on VScode to help with latex code, didn't compile the current file as usual maybe just after I initiate a git repository.

I thought there maybe something wrong in the hash table where file indexes, so I tried reinstall the extension, update the VScode, reboot the computer. But no matter how many times I tried to touch the compile button, or even I created a new file outside the repository, it just didn't work. So I realize maybe it's not problem of me, nor extension or computer. When I want to give up, I intend to open the compiler log and extension log, although I knew the compiler of \LaTeX is so intricate that I can't even understand its output generally.

1.2 explanation

In \LaTeX , a main way for compiler to determine where the main file is is to detect `\begin{document}` `\end{document}`. But when I copy code from a template, to modify some macro-define value, I delete that line by mistake. So compiler didn't treat it as a main file.

1.3 reflect

Such a nonsensical bug took me a whole afternoon, which told me **reading the compiler log and even extension log is necessary** even though sometime the normal log is hard to understand. However **it doesn't means the error especially nonsensical error report is complicated** as well.

BUG2. RISC-V Assembly Calling Convention

2.1 bug itself

In summer holiday I took a open course about introduction to computer architecture. In Proj2, which helps students to be familiar with RISC-V assembly grammar and calling convention. I passed all the local tests, but when facing the eventual test about calling convention I failed.

As a background, calling convention in RISC-V means some registers' value may change after calling function while others mustn't do. This characteristic is implemented by coder instead of hardware itself so coder should store the latter ones at the beginning of the function in stack and load back at the end. I checked all files and make sure I conform the rule, but it just can't pass. I once give up all. But one day after lunch, I decide to read the code again, then I found the absurd bug source.

2.2 explanation

When I change a register's role to represent another thing, I forget to change it at another place, which won't get into an error but will break the calling convention. Consequently when calling the function again, the register's value changed and need to reload the original value.

The whole reason that causes the bug is I have a (I thought good at the beginning) bad habit that I always want to optimize the code. When the habit functions on RISC-V, I always want to use less register to finish the work, which causes that I change what a register represent in the middle after I ascertain the former won't be used later. Not only does this habit cost extra time to ponder, but it is also easy to write wrong code hard to find bug in.

2.3 reflect

This experience have a huge impact on me.

First, as the course's instructor **Professor.Weaver** said, every time you want to optimize the code, you'd better **think about why you should optimize it**, otherwise it doesn't make sense and is completely useless.

Second, I read other people's code after debugging and found that they even tried using all register and comment on almost each line, which I thought was not "elegant". However this won't cause bugs basically. So another lesson for me is **commenting more especially when trying not so secure operations and being elegant is not to optimize without pondering**.

Reflect on all

Typically I seldom get blocked by bugs comparing to times when I had no idea about how to code at all thanks to modern debugger, step by step debugging method and Professor.Wu who taught me the skill on my first computer class. It is because of my reliability on debugger instead of **compiler and debugging which are important as well** that I always be stuck by absurd bugs and stay in helplessness. The bugs above are classical examples.

Other things

L^AT_EX code refer to these things and was compiled on texlive2020 by xelatex.

- [UCB-CS70's given homework template.](#)
- [A free website useful to edit L^AT_EX formula code.](#)

The purpose of writing in English is to adapt to bilingual teaching and to improve my poor English writing skills in preparation for a possible future exchange program.

Thanks for your correcting and grading :).