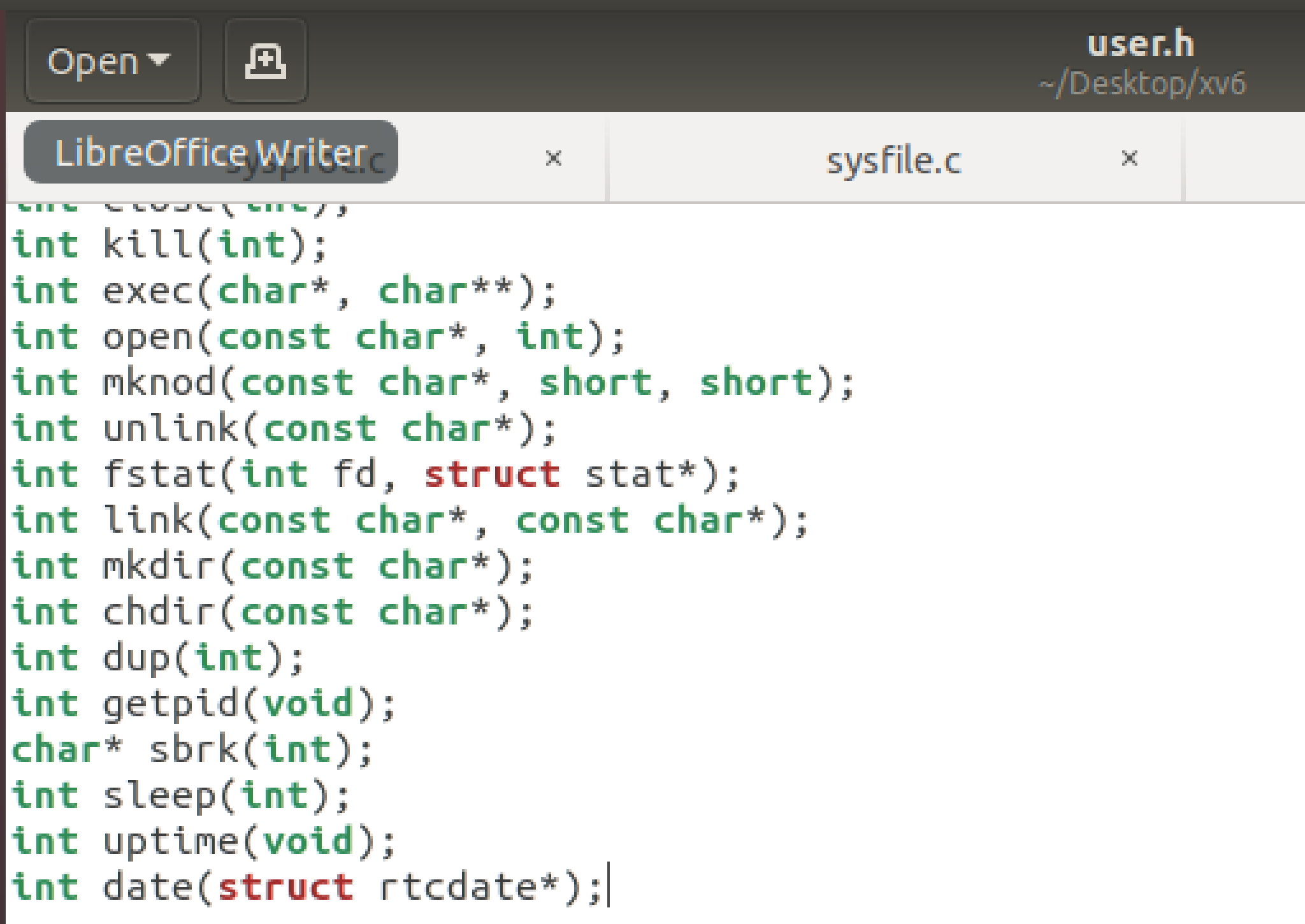
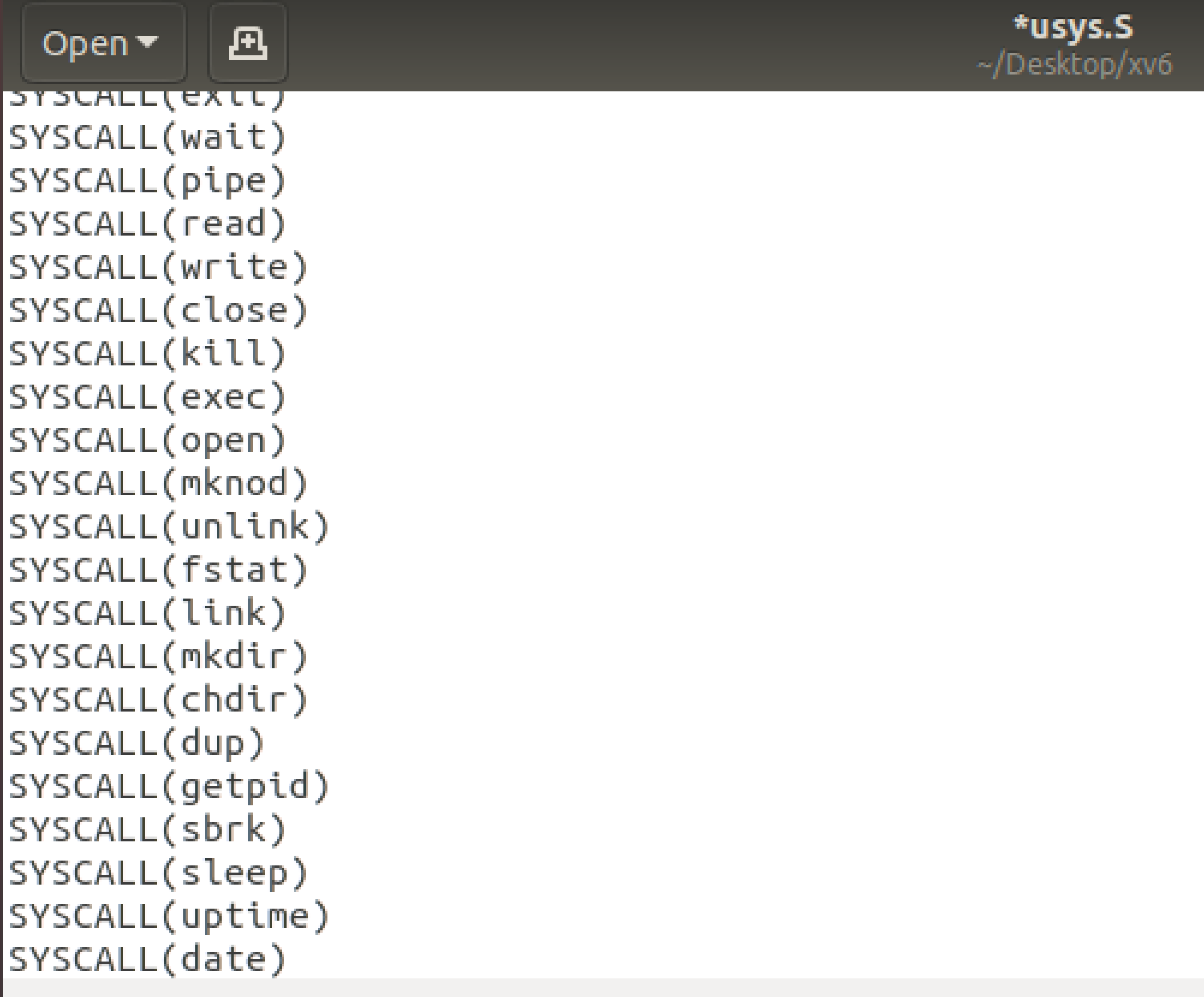
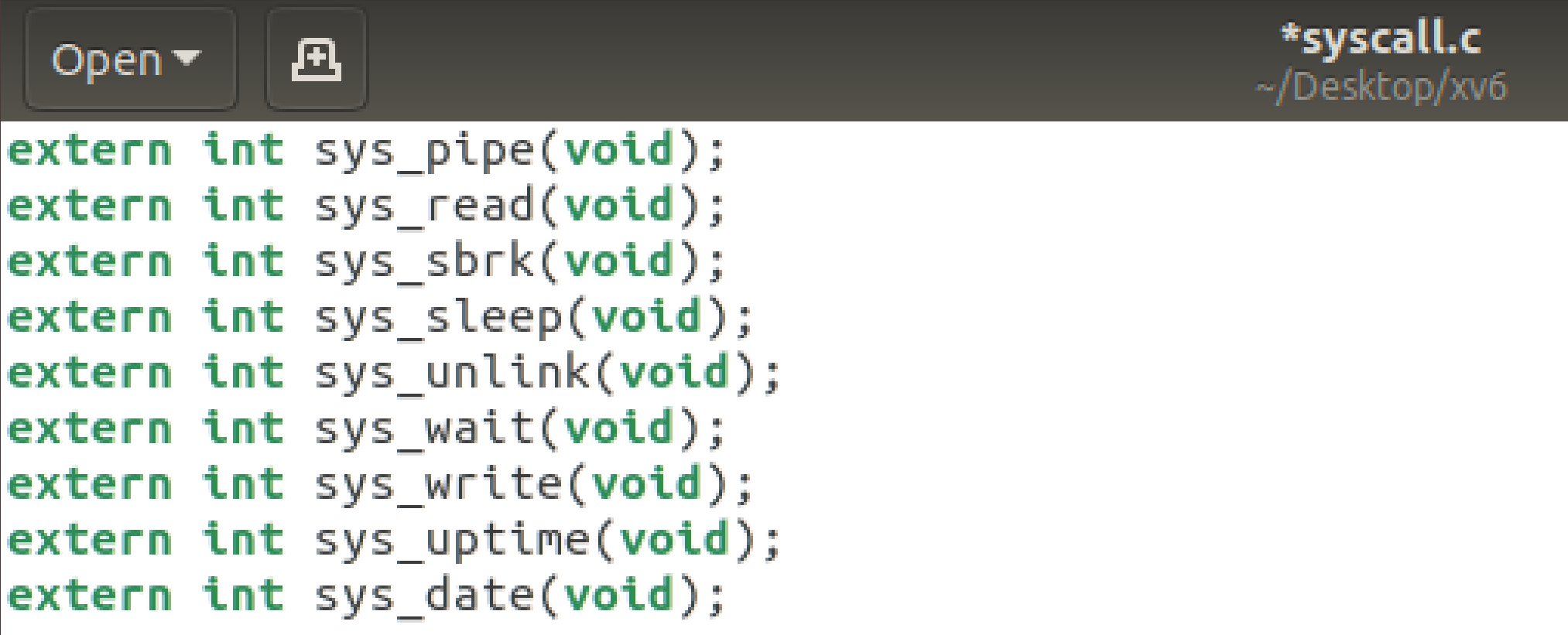
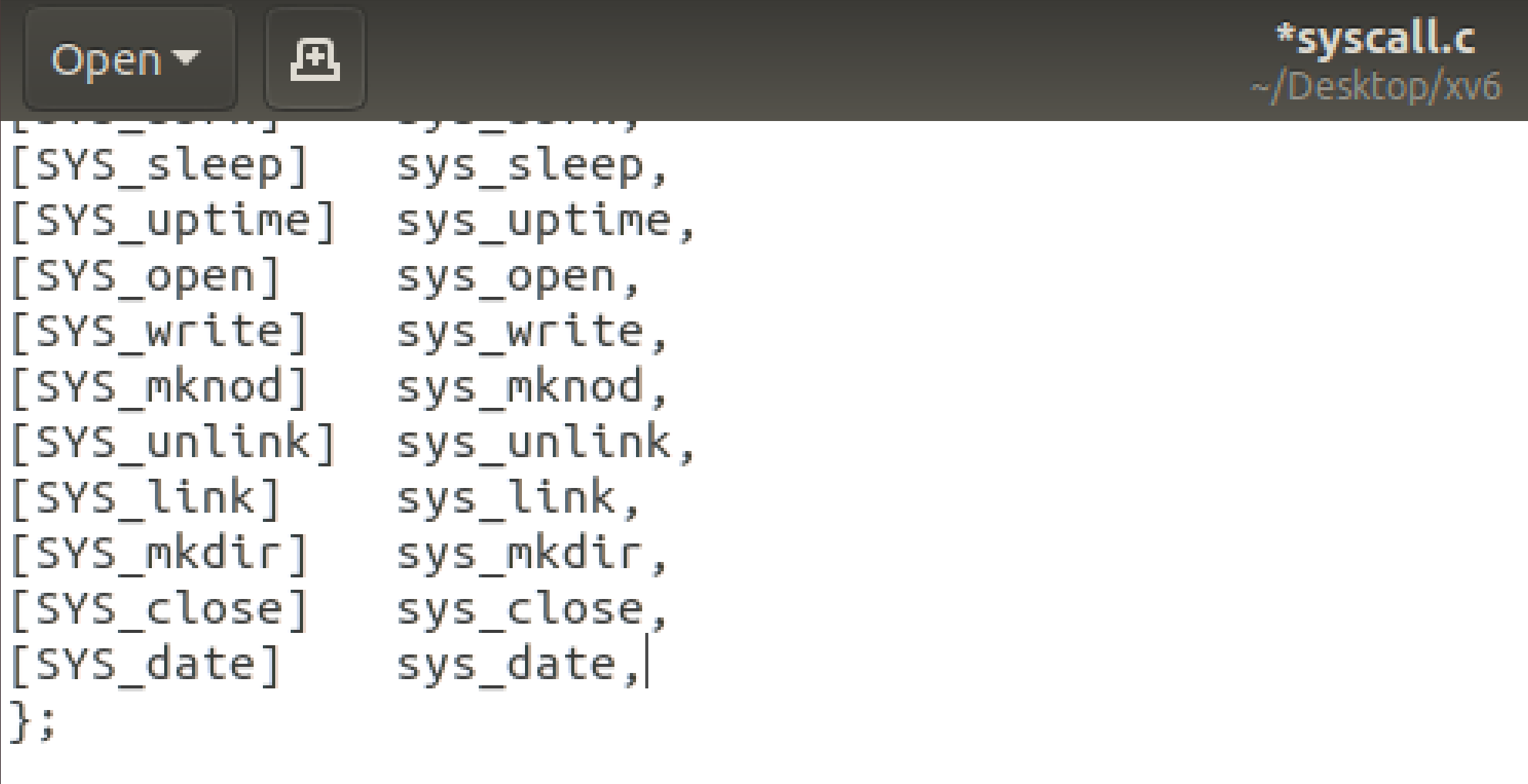
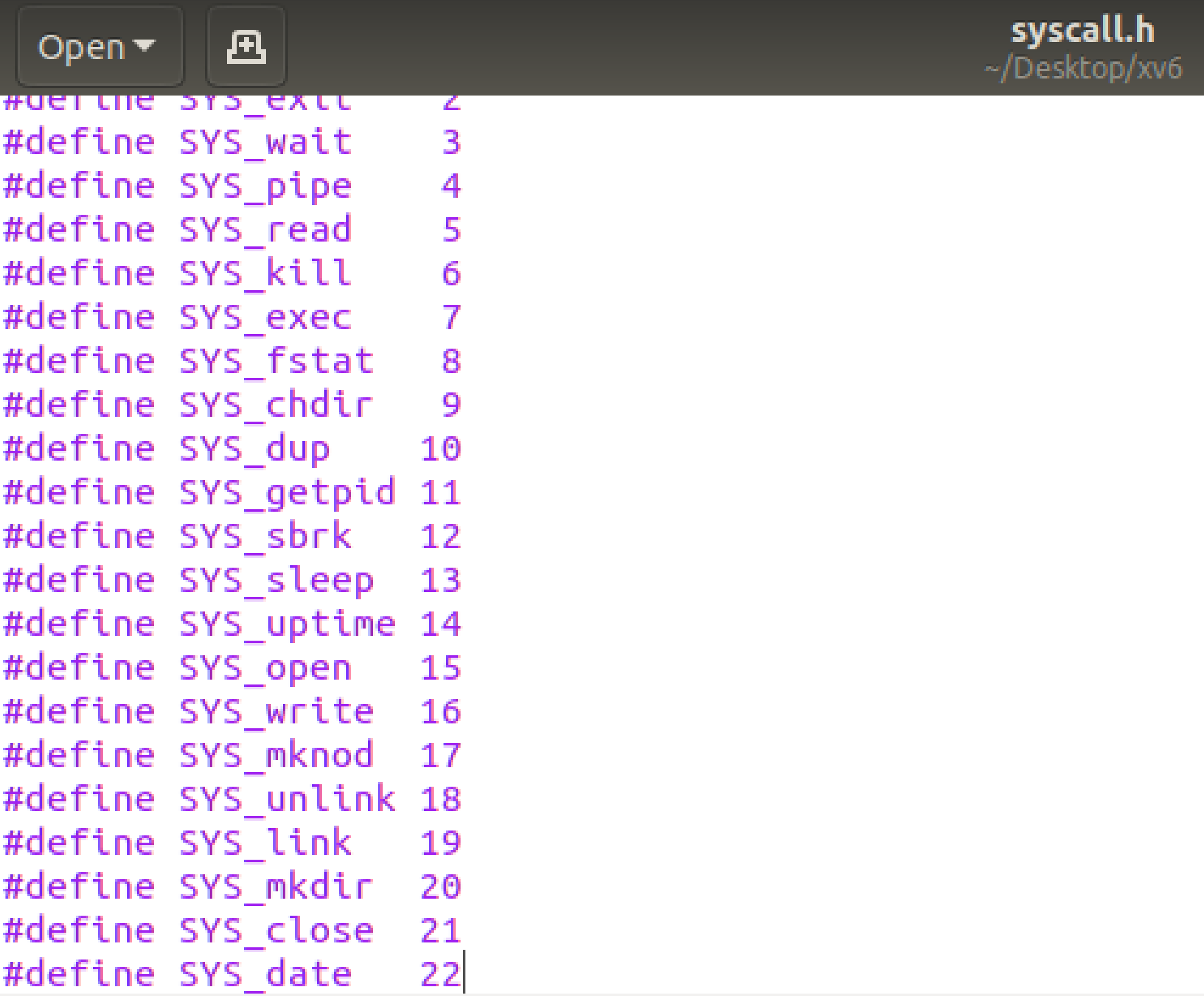
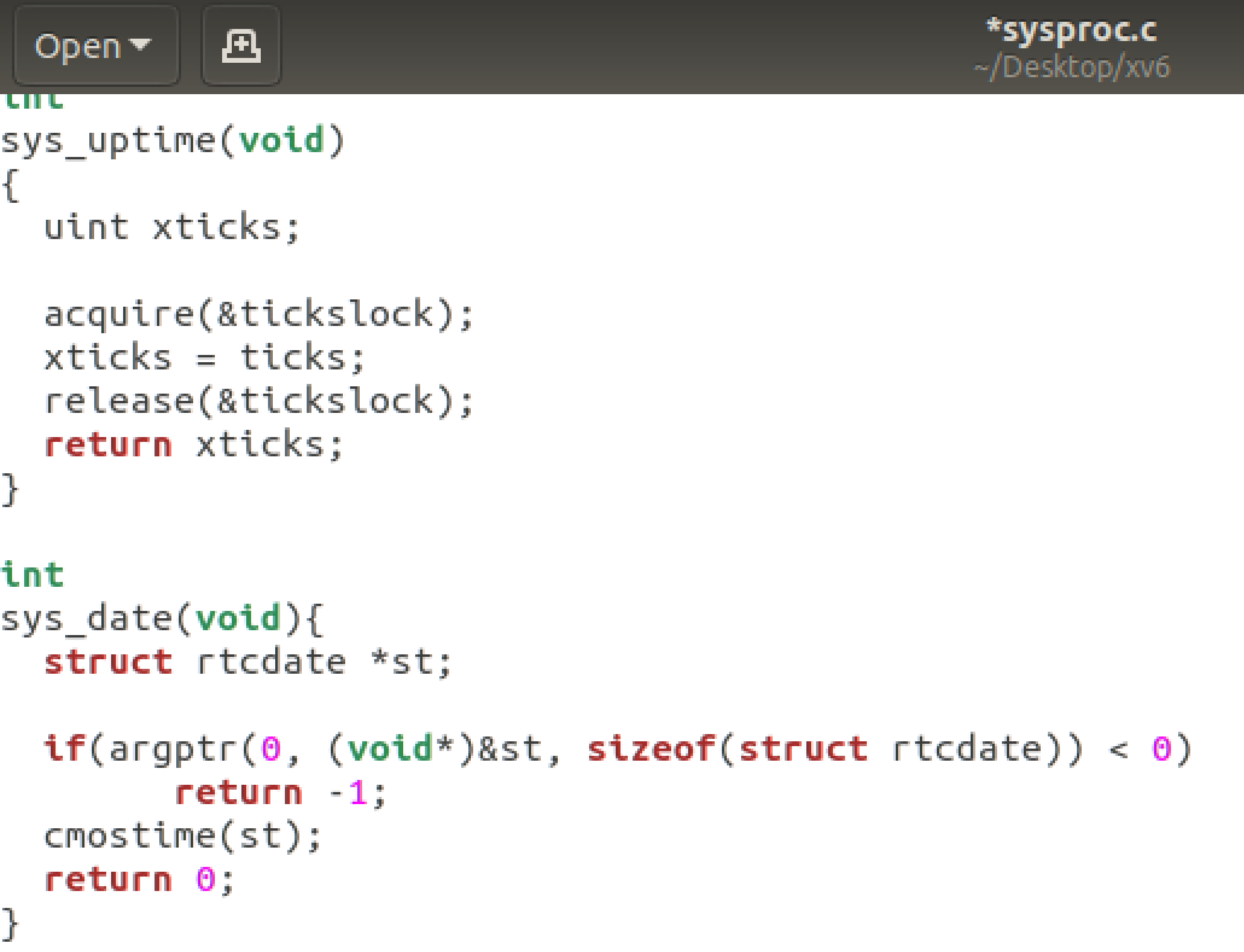
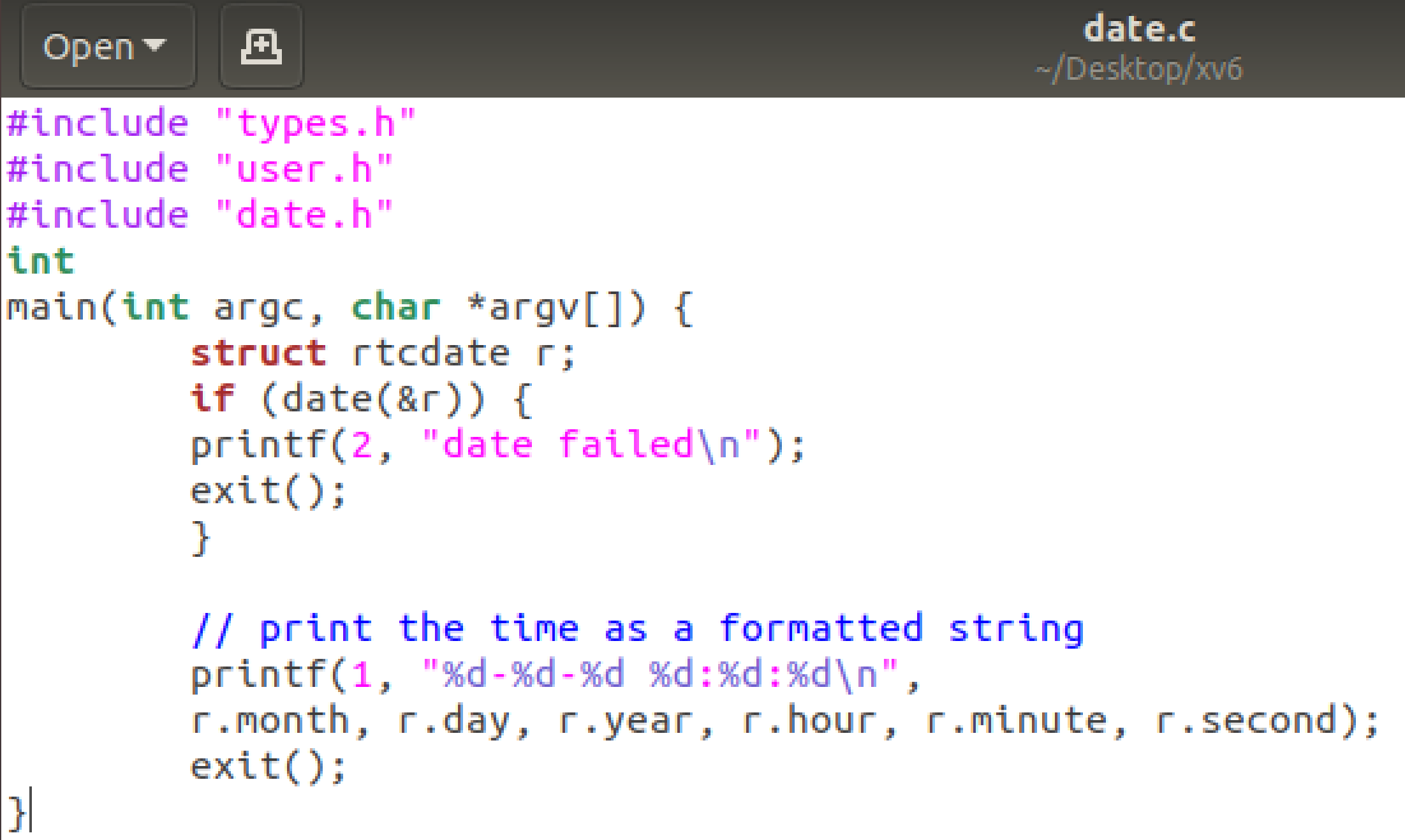
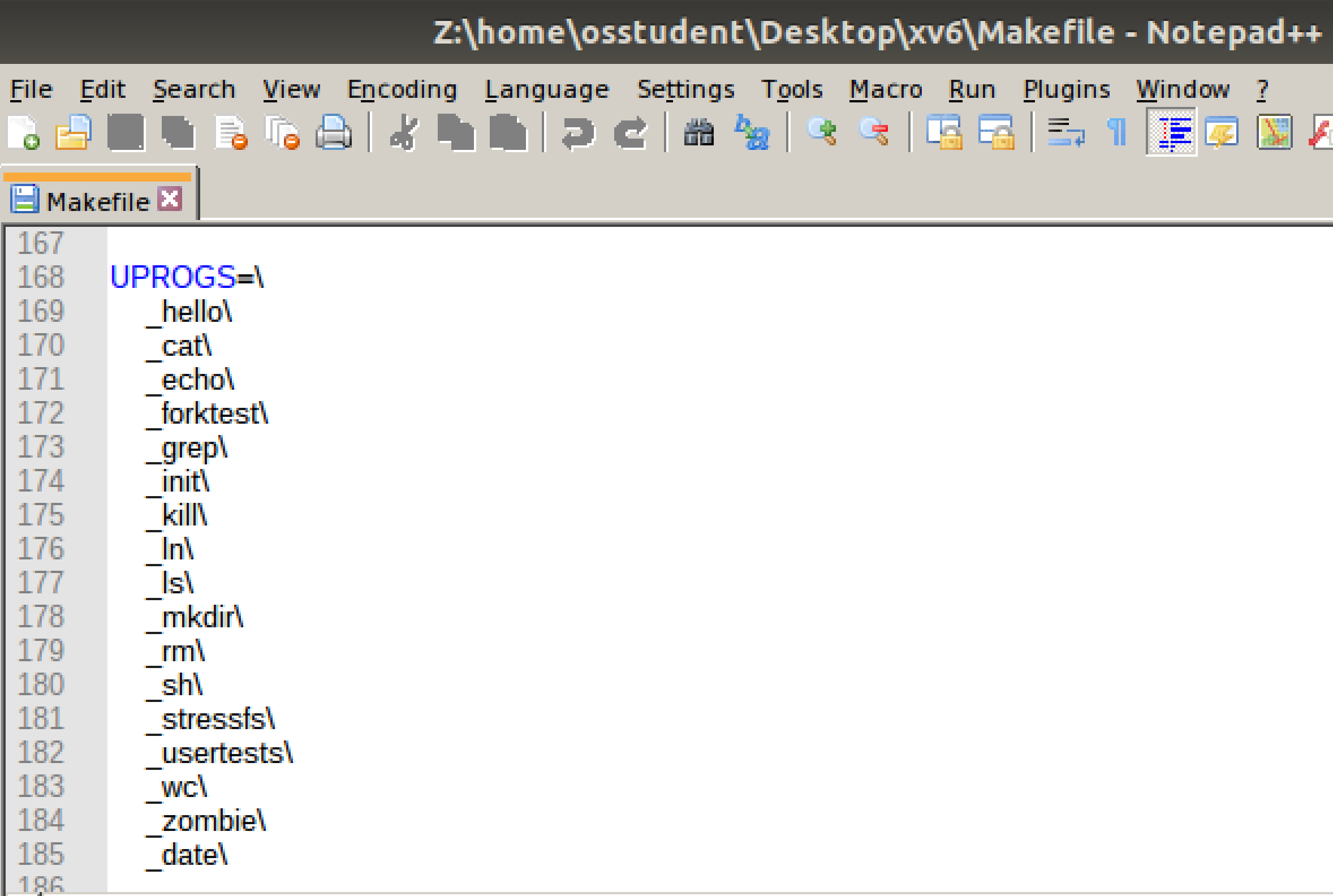
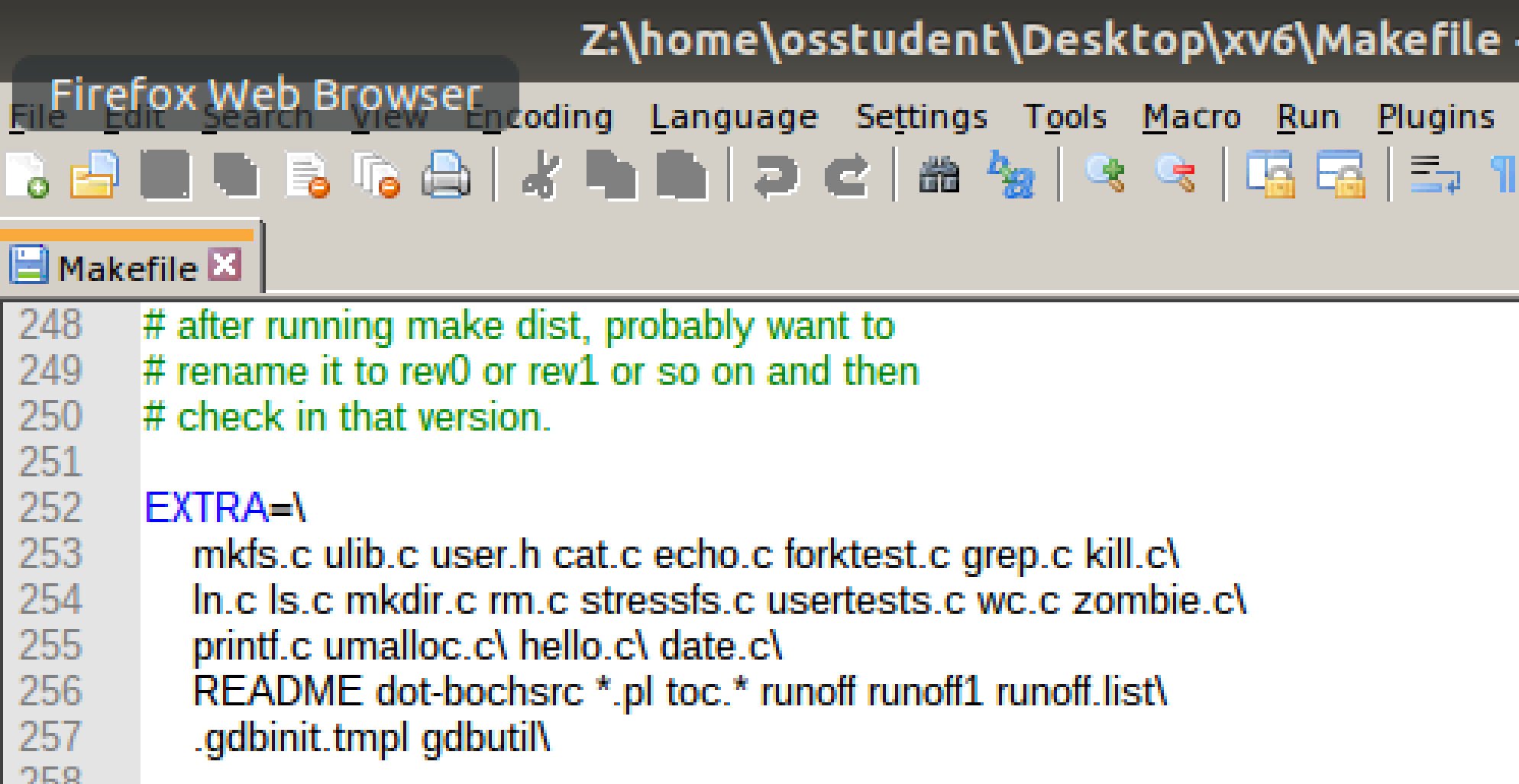
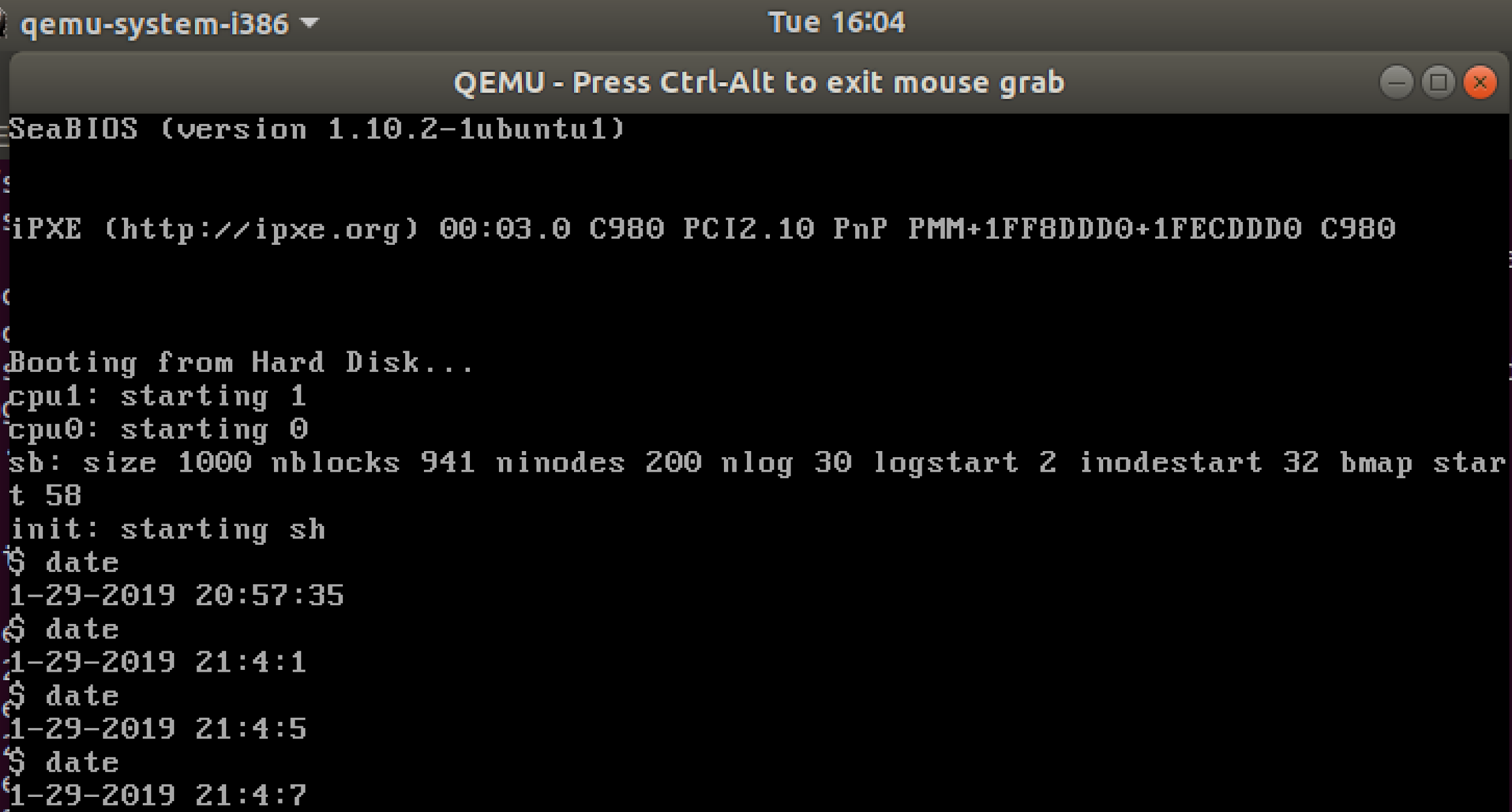
Jaycie Raby, Srinivas Simhan

CIS 450 - Guo

29 January 2019

**Project 2 Report**

1. **The source code of your sys\_date function**
   1. Included in .zip file
2. **A brief explanation for each of the files you have to modify in the process of creating your date system call.**
   1. **user.h**
      1. Added int date(void);
      2. 
   2. **usys.s**
      1. Added SYSCALL(date)
      2. 
   3. **syscall.c**
      1. Added extern int sys\_date(void);
      2. 
      3. Added [SYS\_date] sys\_date,
      4. 
   4. **syscall.h**
      1. Added #define SYS\_date 22
      2. 
   5. **sysproc.c**
      1. Added int sys\_date(void) function (as shown below)
      2. 
   6. **date.c is created**
      1. 
      2. **Edit UPROGS=\ in MakeFile**
         1. 
      3. **Edit EXTRA=\ in MakeFile**
         1. 
3. **A screenshot date is working**
   1. 
4. **Brief reflection on what you learned (> 100 words)**
   1. Jaycie Raby
      1. During this assignment, I learned a lot more about manipulating files within Ubuntu, in a text editor and through the command lines.
      2. I had issues in the sysproc.c file with the sys\_date(void) function not working, so having to go through the process of determining the issue and how to fix it was very helpful. Unlike void functions in C++, C requires you to have a return 0 function, regardless of it being void.
      3. Another issue was that in the user.h file, I was making the sys\_date a have voided parameters, when it really required having the struct rcdate\*.
      4. In conclusion, I felt it was very helpful to do another project using xv6, as it really solidifies the meaning and understanding of not only how to manipulate files, but why the operating systems NEEDS you to change them. Also, encountering all of these errors prepares you for how to deal with similar issues going forward!
   2. Srinivas Simhan
      1. This assignment provided me the opportunity to refocus on fundamentals on modifying files through command prompt and text editors on Ubuntu.
      2. An issue we faced was using the “make qemu” command properly. We kept using the default “date” command that was the system Linux command, and not the date command we thought it was. After redefining the date function’s parameters, we were able to format the output so that it came out in current UTC time instead of in current EST time.
      3. This assignment was a practical method of emphasizing the importance of understanding how files/directories are organized in most operating systems - as it is the same methods of ‘C’ code that are used for Ubuntu and Windows to my knowledge. I’m personally looking forward to the next assignment!