Exercise 3

已開始： 9月6日 16:55

測驗說明

GDB

問題 1 **100 分數**

For this exercise, you will gain familiarity with the command line debugging tool gdb.

- On CLEAR (ssh **netID@ssh.clear.rice.edu, (mailto:netID@ssh.clear.rice.edu,)** replacing netID with your own), make a folder called factorial, and change your working directory to that folder
  - Copy the code of **this (https://canvas.rice.edu/courses/59111/files/4776048?wrap=1) ↓ (https://canvas.rice.edu/courses/59111/files/4776048/download?download_frd=1)** file into a new file on CLEAR in the factorial folder
- Compile the c file with the -g (gdb) flag:
  - ```
    gcc -g factorial.c
    ```
- Use ls to view that there is an executable, which by default is called a.out
- Now run it:
  - ```
    ./a.out
    ```
- It should ask you for a number to compute the factorial for, give it a number and notice that the output is incorrect (zero)
- Now we will debug the program with gdb by running the following command:
  - ```
    gdb a.out
    ```
- You are in gdb, and can now play around. Start by entering the following command, which will show you the code
  - ```
    list
    ```
- It only shows you part of the code (10 lines), you can hit enter again (which repeats the previous command, list) to see more of the code
- Let's run it inside gdb:
  - ```
    run
    ```
    ```

- Notice we are still getting zero, so let's inspect what is happening with the program using gdb's features - start by setting a break point at main:
  - `b main`
- For sanity, let's list the breakpoints we've enabled:
  - `info break`
- Now let's run again, which will stop at the breakpoint that we enabled:
  - `run`
- Now we want to watch some variables to see how they change - let's watch f (factorial) and n (number to compute the factorial for)
  - `watch f`
  - `watch n`
- Do info break again - notice that now f and n are included
- Hit the following to continue running the program:
  - `c`
- Enter 4, and then hit enter again multiple times to notice how f and n change throughout the program
- Notice that eventually f goes from 24 to 0, this is not good!
- Let's verify f really is 0 by using print:
  - `print f`
- Now figure out how to change the code so that it correctly computes the factorial, and submit a screenshot displaying the output of your corrected factorial code working for two different inputs

上傳 選擇檔案