

Q1.

```
loulan@ubuntu:~$ mkdir ~/11810935
loulan@ubuntu:~$ ls
11810935  Documents  Music      Public     Videos
Desktop  Downloads  Pictures   Templates
```

Q2.

```
loulan@ubuntu:~$ ls -la ~
total 76
drwxr-xr-x 15 loulan loulan 4096 Jan 20 01:09 .
drwxr-xr-x  3 root  root  4096 Jan 13 06:39 ..
drwxrwxr-x  2 loulan loulan 4096 Jan 20 01:09 11810935
-rw-r--r--  1 loulan loulan   8 Jan 13 19:11 .bash_history
-rw-r--r--  1 loulan loulan  220 Jan 13 06:39 .bash_logout
-rw-r--r--  1 loulan loulan 3771 Jan 13 06:39 .bashrc
drwxr-xr-x 10 loulan loulan 4096 Jan 13 06:48 .cache
drwx----- 10 loulan loulan 4096 Jan 13 06:49 .config
drwxr-xr-x  2 loulan loulan 4096 Jan 13 06:43 Desktop
drwxr-xr-x  2 loulan loulan 4096 Jan 13 06:43 Documents
drwxr-xr-x  2 loulan loulan 4096 Jan 13 06:43 Downloads
drwx-----  3 loulan loulan 4096 Jan 13 06:43 .gnupg
drwxr-xr-x  3 loulan loulan 4096 Jan 13 06:43 .local
drwxr-xr-x  2 loulan loulan 4096 Jan 13 06:43 Music
drwxr-xr-x  2 loulan loulan 4096 Jan 13 06:43 Pictures
-rw-r--r--  1 loulan loulan  807 Jan 13 06:39 .profile
drwxr-xr-x  2 loulan loulan 4096 Jan 13 06:43 Public
drwxr-xr-x  2 loulan loulan 4096 Jan 13 06:43 Templates
drwxr-xr-x  2 loulan loulan 4096 Jan 13 06:43 Videos
```

Q3.

```
loulan@ubuntu:~$ cd ~/11810935
loulan@ubuntu:~/11810935$
```

Q4.

```
loulan@ubuntu:~/11810935$ man grep
```

```

GREP(1)
User Commands
GREP(1)

NAME
    grep, egrep, fgrep, rgrep - print lines that match patterns

SYNOPSIS
    grep [OPTION...] PATTERNS [FILE...]
    grep [OPTION...] -e PATTERNS ... [FILE...]
    grep [OPTION...] -f PATTERN_FILE ... [FILE...]

DESCRIPTION
    grep searches for PATTERNS in each FILE. PATTERNS is one or more patterns separated by newline characters, and grep prints each line that matches a pattern. Typically PATTERNS should be quoted when grep is used in a shell command.

    A FILE of "-" stands for standard input. If no FILE is given, recursive searches examine the working directory, and nonrecursive searches read standard input.

    In addition, the variant programs egrep, fgrep and rgrep are the same as grep -E, grep -F, and grep -r, respectively. These variants are deprecated, but are provided for backward compatibility.

OPTIONS
    Generic Program Information
    --help Output a usage message and exit.

    -V, --version
        Output the version number of grep and exit.

```

Manual page grep(1) line 1 (press h for help or q to quit)

Q5.

```

loulan@ubuntu:~$ sudo mv ~/11810935 /home
loulan@ubuntu:~$ ls /home
11810935  loulan

```

Q6.

```

loulan@ubuntu:~$ sudo rm -r /home/11810935
loulan@ubuntu:~$ ls /home
loulan

```

Q7.

```
loulan@ubuntu:~$ sudo cp /etc/apt/sources.list /etc/apt/sources.list.bak
```

Q8.

```
loulan@ubuntu:~$ cat /etc/shells
# /etc/shells: valid login shells
/bin/sh
/bin/bash
/usr/bin/bash
/bin/rbash
/usr/bin/rbash
/bin/dash
/usr/bin/dash
```

Q9.

```
loulan@ubuntu:~$ cat /etc/shells | grep bash
/bin/bash
/usr/bin/bash
/bin/rbash
/usr/bin/rbash
```

Q10.

Open two terminals and show their PIDs:

```
loulan@ubuntu:~$ ps
  PID TTY          TIME CMD
 1889 pts/0    00:00:00 ba
 1906 pts/0    00:00:00 ps
loulan@ubuntu:~$
```

```
loulan@ubuntu:~$ ps
  PID TTY          TIME CMD
 1920 pts/1    00:00:00 bash
 1935 pts/1    00:00:00 ps
loulan@ubuntu:~$
```

Kill one of them:

```
loulan@ubuntu:~$ ps
  PID TTY          TIME CMD
 1889 pts/0    00:00:00 bash
 1906 pts/0    00:00:00 ps
loulan@ubuntu:~$ kill -9 1920
```

```
  PID TTY          TIME CMD
 1920 pts/1    00:00:00 bash
 1935 pts/1    00:00:00 ps
```



```
loulan@ubuntu:~$ kill -9 1920
loulan@ubuntu:~$
```

Q11.

1) the source code:

Create file opt.c:

```
loulan@ubuntu:~$ touch opt.c
loulan@ubuntu:~$ ls
Desktop  Downloads  opt.c      Public  Videos
Documents Music      Pictures  Templates
```

Edit file opt.c:

```
loulan@ubuntu:~$ vim opt.c
```

```
#include<stdio.h>

int main(){
    int x = 0;
    x += 1;
    x += 1;
    x += 1;
    printf("%d\n", x);
    return 0;
}
```

2) command:

```
loulan@ubuntu:~$ gcc -S opt.c -O0 -o opt0.s
loulan@ubuntu:~$ gcc -S opt.c -O1 -o opt1.s
```

3) result:

Under O0:

```

loulan@ubuntu:~$ cat opt0.s
        .file      "opt.c"
        .text
        .section   .rodata
.LC0:
        .string   "%d\n"
        .text
        .globl    main
        .type      main, @function
main:
.LFB0:
        .cfi_startproc
        endbr64
        pushq     %rbp
        .cfi_def_cfa_offset 16
        .cfi_offset 6, -16
        movq      %rsp, %rbp
        .cfi_def_cfa_register 6
        subq      $16, %rsp
        movl      $0, -4(%rbp)
        addl      $1, -4(%rbp)
        addl      $1, -4(%rbp)
        addl      $1, -4(%rbp)
        movl      -4(%rbp), %eax
        movl      %eax, %esi
        leaq      .LC0(%rip), %rdi
        movl      $0, %eax
        call      printf@PLT
        movl      $0, %eax
        leave
        .cfi_def_cfa 7, 8
        ret
        .cfi_endproc
.LFE0:
        .size      main, .-main
        .ident     "GCC: (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0"
        .section   .note.GNU-stack,"",@progbits
        .section   .note.gnu.property,"a"
        .align     8
        .long      1f - 0f
        .long      4f - 1f
        .long      5
0:
        .string    "GNU"
1:
        .align     8
        .long      0xc0000002
        .long      3f - 2f
2:
        .long      0x3
3:
        .align     8
4:

```

Under O1:

```
loulan@ubuntu:~$ cat opt1.s
.file "opt.c"
.text
.section .rodata.str1.1,"aMS",@progbits,1
.LC0:
.string "%d\n"
.text
.globl main
.type main, @function
main:
.LFB23:
.cfi_startproc
endbr64
subq $8, %rsp
.cfi_def_cfa_offset 16
movl $3, %edx
leaq .LC0(%rip), %rsi
movl $1, %edi
movl $0, %eax
call __printf_chk@PLT
movl $0, %eax
addq $8, %rsp
.cfi_def_cfa_offset 8
ret
.cfi_endproc
.LFE23:
.size main, .-main
.ident "GCC: (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0"
.section .note.GNU-stack,"",@progbits
.section .note.gnu.property,"a"
.align 8
.long 1f - 0f
.long 4f - 1f
.long 5
0:
.string "GNU"
1:
.align 8
.long 0xc0000002
.long 3f - 2f
2:
.long 0x3
3:
.align 8
4:
```

4) difference:

oo 没有进行任何的优化，是默认的编译选项。

O1 会对程序进行一定程度的优化，包括但不限于缩减对应编译后代码的大小。比如，在 **opt0.s** 的 **.LFB0** 代码段的 **subq** 到 **leaq** 命令之间有多行 **movl** 与 **addl** 命令，但在 **opt1.s** 对应的 **.LFB23** 代码段却只有一条 **movl** 命令。但编译器不会执行需要占用大量时间的优化。