



PHYSICS 2T

C Programming under Linux

CLASS TEST

Candidates should answer:

- Question 1 [20 marks];
 - Question 2 [20 marks];
 - Question 3 [20 marks].
-
- You must not leave the examination room in the first 30 minutes and not within the last 15 minutes of the examination.
-

Question:

1.

- a) An ascii formatted file called names.txt contains a single column of 100 surnames that are a maximum of 12 characters in length. Write a short C-program containing a loop that reads in each name and prints it to the terminal.

[6]

- b) Next, to store the 100 surnames in the program, add some code to declare and fill a suitable array. Determine the length of the longest name read.

[4]

- c) Now suppose that the code you have written is one of four C source files for a larger project. The full list of source files is: source1.c, source2.c, source3.c and source4.c. Write down a single Linux command that would compile these source files, link them to the maths library and produce an executable called 'simulation'

[4]

- d) Create a Linux makefile based on your answers to part a) so that linking is performed in a separate step and that the source files are only compiled as and when necessary. Make appropriate use of automatic variables. Print to the screen at each step what the make process is about to do.

[6]

Continued overleaf

2.

a) Write down a shell script that prints the number of command line arguments and displays their contents to the screen.

[4]

b) Write a script that checks to see if the entries in a list are files or directories. If they are files, check to see if they are executable and if so, execute them. If the file is not executable, exit with a non-zero status. If they are directories, change directory into them and print that you have done so, before continuing onto the next argument.

[8]

c) Explain what the grep command does. How could you use grep to identify any bash scripts in the current directory? Write the command down.

[3]

d) Explain what the make automatic variables % and ^ mean.

[2]

e) Explain the meaning of the -g, -Wall and -O2 switches to gcc.

[3]

Continued overleaf

3.

Examine the C code below and answer the following questions:

- a) The `scanf` function is used (correctly) in line-14: Write down code using the `fgets` and `sscanf` functions that could replace this line. [4]
- b) Explain the use of the “==”, “&&”, “||” and “++” operators in line-15. [4]
- c) How must the program be modified to write a binary output file? [4]
- d) The program should start with a block of comments that includes a description of the program. How are comments included? Write down a suitable program description, including a description of the key variables and any limitations on the program’s use. [4]
- e) The `list` and `sum` structs are initialized using the "anonymous" syntax. How would you modify the initialization of the `sum` struct to explicitly specify each element assigned? [2]
- f) The `list` structure is declared in the file scope: explain what this means for the allocation of its memory, and its visibility to other functions that might be included in this file at a later date. [2]

```
#include <stdio.h>

struct complex {
    int re;
    int im;
} list[5] = {{0,0},{0,0},{0,0},{0,0},{0,0}};

int main(void)
{
    FILE *out_file_ptr;
    int i = 0;

    while (1) {
        printf("Enter real and imaginary magnitudes or 0 0 to end: ");
        scanf("%d %d", &list[i].re, &list[i].im);
        if ((list[i].re == 0 && list[i].im == 0) || ++i == 5)
            break;
    }

    struct complex sum = {0,0};
    for (i=0; i<5; i++) {
        sum.re += list[i].re;
        sum.im += list[i].im;
        printf("%d + %di\n", list[i].re, list[i].im);
    }
    printf("Sum is %d + %di\n",sum.re, sum.im);

    out_file_ptr = fopen("output.dat","a+");
    fprintf(out_file_ptr,"%d %d\n",sum.re,sum.im);
    fclose(out_file_ptr);

    return 0;
}
```