

Home Exam 1 for PG3400 - Basic C Programming

Vamsidhar Reddy Gaddam

September 2014

Abstract

In the following document, you will find the question, hints and instructions for the first Home Exam for PG3400. This exercise helps you in getting started with basic programming including using conditional statements, pointers, memory allocations and deallocation.

1 Instructions

- You are encouraged to discuss about the problem and possible solutions, but your code is expected to be independent.
- Discussions on the forum might be particularly useful since I follow them too.
- The deadline is 5th of October and it is enforced by the system.
- Any hint of plagiarism will be heavily penalized.
- The style of the code will also be considered in grading.

2 Number input

In the course we saw how to input from the terminal using `scanf()`. But You can use the following snippet to read from the file. Here `FILENAME` is a string containing the filename.

```
FILE* f = fopen (FILENAME, "r");
int i = 0;
fscanf (f, "%d", &i);
while (!feof (file))
{
    fscanf (f, "%d", &i);
}
fclose (f);
```

Write a program that takes a filename as input and reads the integers from the file until the end of file. This program should store the numbers in an array and also the count in a variable. The count could vary from 2 integers to 200000 integers.

Example : `./myprogram 1.txt`

Execution of this must read all the integers in 1.txt

BONUS: For dynamic memory allocation

3 Sorting

Once the numbers are read into the memory, you should sort them and have them in the same array.

Recommended sort: Bubble sort

4 Search

After the sorting of the integers is performed. The program will output the following message :

Please input an integer for search:

Then the output should be :

- if the integer is found, the position of the integer in the sorted array
- else, inform the user that the integer is not in the array
- if the input is 0 - quit the program

Recommended Search: *Binary search*

BONUS: If you can find the position of the searched integer in the original file. You might want to store the index somewhere when you sort!