INDIAN STATISTICAL INSTITUTE

Laboratory Test I

M. Tech (CS) - I Year, 2015-2016 (Semester - I)

Data and File Structures Laboratory

Date: 31.07.2015 Total Marks: 130 + 20 = 150 (20 marks for good programming habits)

Time: 3 Hours

Note: Follow the file naming convention strictly as mentioned.

You are not allowed to connect and browse the internet during the test. No books and e-books are allowed. Any instance of malpractice would be dealt with sternly.

(Q1) Write a C program (Name your program `labtest1-sort-cs15XX.c` where `XX` is your roll number) that

(a) takes as input three integers — n, x and z from the user. [4]

(b) reads n integers into an array $\mathcal{A}$ from the user. [8]

(c) sorts the values in the array $\mathcal{A}$ in nondecreasing order. (You can reuse the code developed in the class). [8]

(d) performs *binary search* to find whether x lies in $\mathcal{A}$. (Binary Search is described in Algorithm 1 for you.) Count the number of comparisons you make. [20+5]

(e) finds whether there exists two integers x_1 and x_2 in the array $\mathcal{A}$ such that $x_1 + x_2 = z$. Count the number of comparisons you make. [20+5]

Total marks: [4+8+8+25+25=70]

Algorithm 1: Binary search

Input: An array $\mathcal{A}$ of n elements sorted in nondecreasing order and an element x;
Output: The index j if x equals $\mathcal{A}[j]$, $0 \leq j \leq n-1$, and -1, otherwise

```
low = 1; high = n − 1; j = −1;
while ( ( low ≤ high ) && (j == −1) ) do
    mid = ⌊( low + high )/2⌋;
    if (x == A[ mid ]) then
        j = mid ;
    else
        if (x < A[ mid ]) then
            high = mid - 1;
        else
            low = mid + 1;
return j;
```

(Q2) The `sort` command sorts the lines of the input in the order specified by the options. For example, `sort filename` sorts the lines of the file `filename` by (default) lexicographic order. Read about the options for `sort` in the manual page, using the command `man sort`. Note that more than one options can be given to `sort`. For example, to sort in reverse order (`-r`) and ignoring the case (`-f`), you can use `sort -fr`.

You have already encountered the `grep` command in your assignment. The command `grep "word" filename` outputs all lines containing the string `word` in the file `filename`. For more specific matching, the `^` (beginning of the line) and `$` (end of the line) operators can be used. For example, the command `grep "word$" filename` would output all the lines ending with the string `word` in the file `filename`. For parts (b) - (e), create a file named `labtest1-2-commands-cs15XX` and write your commands in that file. Write the question number starting with a # in one line and write the commands for that question in the following lines without any #.

(a) Write a C program which prints 1000 numbers, one in each line, according to the following scheme: start with any number a_1 such that $0 \leq a_1 \leq 99$. If the i-th number is a_i, then next number a_{i+1} would be $2a_i \bmod 100$, that is, the remainder obtained by dividing $2n$ by 100. For your reference, the mod operator in C is % (example: you write `z = x % y` in C to assign `x mod y` to `z`). Name your program `labtest1-2a-numbers-cs15XX.c`, where `XX` is your roll number. Compile the program and name the executable `numbers`. [12]

(b) Write the command to run your C program and save the output of the program to a file named `numbers.txt`. [2]

(c) Write a Linux command to sort the file `numbers.txt` in descending order of the numbers. [8]

(d) Write a Linux command, or a combination of commands (but in a single line), to output the number of unique numbers in the file `numbers.txt`. [10]

(e) Write a Linux command to output only the numbers that are divisible by 10 from the file `numbers.txt`. [8]

(f) Write a Linux command to output all the numbers that are not divisible by 5 from the file `numbers.txt`. [12]

(g) Write one or more Linux commands to swap the names of two files `file1` and `file2`. In other words, after running your command(s), the original content of `file1` should be in `file2` and vice versa. Please note that no intermediate file should remain. Try to perform this using least number of commands. [8]

Total marks: [12+2+8+10+8+12+8=60]

Submission instruction: Submit all your source codes to the directory `2015/labtest1/cs15XX` under the home directory of the user `pdslab`. For example, if you are copying a file `labtest1-sort-cs15XX.c`, you should use the command

```
cp labtest1-sort-cs15XX.c ~pdslab/2015/labtest1/cs15XX
```

where `XX` is your roll number.