



United International University
Department of CSE
CSE 1112: Structured Programming Language Lab
Final Examination | Spring 2023 | Set A
Duration: 9:00 - 11:00
Time: 1 Hour 20 Minutes | Full Marks: 25

Class ID:

Name:

Note: Figures on the right-hand side indicate full marks.

1. Write a program that checks whether a number is a “Superprime” number or not. A “Superprime” number is a number that has the following properties: **13 Marks**

- The number is prime
- The sum of all the digits in that number is also a prime number

For example, the number “223” is a “Superprime” because it is a prime number, and also, the sum of its digits is $2 + 2 + 3 = 7$, which is also a prime number. However, the number “149” is not a “Superprime”, because although it is a prime number, the sum of its digits ($1 + 4 + 9 = 14$) is not prime.

Write a program that takes two numbers a and b as input, and then prints all superprime numbers in the range from a to b (including a and b).

For that purpose, you have to write the following functions:

- `int is_prime(int x)`: This function returns 1 (or nonzero/true value) if the integer x is prime and 0 otherwise
- `int digit_sum(unsigned int x)`: This function takes an integer as input and returns the sum of all the digits.

You must write this function by using recursion.

- `int is_superprime(int x)`: This function takes an integer as input and returns 1 (or nonzero/true value) if it is a superprime and 0 otherwise. **You should make function calls to functions (a) and (b) in this function.**

See the following sample input and output:

Sample Input	Sample Output
100 150	101 113 131 137 139
200 250	223 227 229 241

2. Mr Gaben, the legendary programmer, is working on his company's server when he detected that computer viruses exist in his server. However, he does not know which programs running on the server are computer viruses and which ones are legit. If he does not find out which programs are viruses, all of the valuable data on the server will be lost. So Gaben needs your help in finding these viruses.
- In Gaben's server, each program is represented by a structure, containing the following member variables:

```
struct program {
    char name[60];           // name of the program
    int total_memory;        // total memory in bytes that
                             // program has been assigned by OS
    int used_memory;         // total memory in bytes that the
                             // program is using now
};
```

Gaben knows that the programs which are viruses consume the entire memory that is assigned to it (Their `used_memory` is equal to or greater than their `total_memory`), and also, all letters in their name are capital letters.

Now write a program that takes information about all the programs in Gaben's server, and then prints how many programs among them are viruses and their names.

The first line of input contains an integer `N` indicating the number of programs running on Gaben's server. The next lines contain information about the individual programs. Each group of three lines contains the name, total bytes, and used bytes of a program respectively. The name of a program consists of a single word.

Your program should have the following properties:

1. Your program should represent all the programs in Gaben's server using an array of structures of type `struct program`
2. **You need to make the following function and call it in your program to calculate the result -**

`int all_caps(char *input):` returns 1 (or nonzero/true value) if all letters in the string `input` are capital letters and 0 otherwise.

You may add additional functions if required for your implementation of the program

See the following sample input-output for clarification:

12 Marks

Sample Input	Sample Output	Explanation
4 CHROME 17179869184 17235823736 codeblocks 4294967296	2 CHROME FORTNITE	<p>The 4 programs running in Gaben's server are as follows:</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>Name: CHROME Total bytes: 17179869184</p> </div>

124245667 Xeyes 655360 655360 FORTNITE 8589934592 8589934592		<table><tr><td>Used bytes: 17235823736</td></tr><tr><td>Name: codeblocks Total bytes: 4294967296 Used bytes: 124245667</td></tr><tr><td>Name: Xeyes Total bytes: 655360 Used bytes: 655360</td></tr><tr><td>Name: FORTNITE Total bytes: 8589934592 Used bytes: 8589934592</td></tr></table> <p>Among these four programs, programs 'CHROME' and 'FORTNITE' are viruses. That is because their names have all capital letters, and their used bytes are equal to or exceed their total bytes. The program Xeyes is not a virus, because even if its used bytes are equal to its total bytes, all letters in its name are not capitalized.</p>	Used bytes: 17235823736	Name: codeblocks Total bytes: 4294967296 Used bytes: 124245667	Name: Xeyes Total bytes: 655360 Used bytes: 655360	Name: FORTNITE Total bytes: 8589934592 Used bytes: 8589934592
Used bytes: 17235823736						
Name: codeblocks Total bytes: 4294967296 Used bytes: 124245667						
Name: Xeyes Total bytes: 655360 Used bytes: 655360						
Name: FORTNITE Total bytes: 8589934592 Used bytes: 8589934592						