SET A

United International University

Department of Computer Science and Engineering

Final Examination Summer 2023

Course Code: **CSE 1112** Course Title: **Structured Programming Language Laboratory**

Date: September 12, 2023 Time: 09:00 AM – 10:00 AM (1 hour) Full marks: 25

Name: Student ID:

Write down C programs for the following problems in Code Blocks (or any C compiler you prefer), and present the code to your instructor after the time is up. You can make rough calculations in this paper.

Problem 1 (Marks: 12)

In the bustling metropolis of Primeville, a **"Superhero Palindromic Prime"** is a prime number with a secret power: it remains unchanged even when its digits are reversed, just like the identity of a true superhero. For instance, numbers like **131 and 757** possess this extraordinary ability.

Your mission is to create a program that identifies and unveils all the hidden Superhero Palindromic Prime numbers within a given range. Channel your inner superhero and construct your code using the following components:

a. **int is_prime(int x):** This function serves as your prime power detector, taking an integer x as input and returning 1 (or a true value) if x is prime, and 0 otherwise.

b. **int reverse_number(int x):** This function possesses the power of time manipulation, accepting an integer x and returning the number obtained by reversing the order of its digits. **You must write this function using recursion**.

c. **int is_palindromic_prime(int x):** This function takes an integer x as input and returns 1 (or true) if x is a palindromic prime number (retains its form when its digits are reversed and also prime), and 0 otherwise. **You should make function calls to functions (a) and (b) in this function.**

d. **void find_superhero_palindromic_primes(int start, int end):** This function prints all Superhero Palindromic Prime numbers hidden within the range **[start, end].**

Sample Input	Sample Output
Enter lower limit: 10 **Enter upper limit: 400**	**Palindromic prime numbers within the range 10 to 1000 are:** **11** **101** **131** **151** **181** **191** **313** **353** **373** **383**

SET A

Problem 2 (Marks: 13)

Tanjiro Kamado wants to eliminate as many demons as possible to make a better world for us. Every demon in our world can be presented as **an element of an array of structure**. Like,

```
struct demons {
    char name[60]; // Name of a demon
    int power; // Power of a demon
};
```

But he needs to fulfill two conditions before eliminating a demon. The conditions are:

- **a. A demon has power multiple of 5, and**
- **b. A demon's name contains only English alphabets.**

Tanjiro can only eliminate a demon who has power multiple of 5 and who has only English alphabets in his name. Like, if a demon has “Kute” as a name and power as 10 then Tanjiro can eliminate that demon. But if a demon has a name “Ku+e” as a name or power as 9 then Tanjiro can not eliminate that demon.

So your task is to find how many demons our Tanjiro can eliminate. Your implementation must have these two functions:

a) **int onlyAlphabets(char *input)** : This function returns 1 if the demon's name has only alphabets as letters (they can be capital or small letters) or return 0 otherwise.

b) **int Multiple_of_5(int x)** : This function returns 1 if the demon’s power is multiple of 5. Returns 0 otherwise.

Sample Input	Sample Output
5 Kute 15 Ku+e 6 sMiley 10 ??x?? 35 PenTagoN 501	2

SET B

United International University

Department of Computer Science and Engineering

Final Examination Summer 2023

Course Code: **CSE 1112** Course Title: **Structured Programming Language Laboratory**

Date: September 12, 2023 Time: 09:00 AM – 10:00 AM (1 hour) Full marks: 25

Name: Student ID:

Write down C programs for the following problems in Code Blocks (or any C compiler you prefer), and present the code to your instructor after the time is up. You can make rough calculations in this paper.

Problem 1 (Marks: 12)

In the mystical realm of Numericopolis, a **"Superhero Fibonacci Palindrome"** is a number with an exceptional fusion of powers: it's **not only a palindrome but also a valiant member of the Fibonacci league**. The Fibonacci league is an assembly where each number is a union of the strengths of its two predecessors, starting from the dynamic duo of 0 and 1. As their journey unfolds, they summon forth numbers like 0, 1, 1, 2, 3, 5, 8, 13, and 21, each a hero in its own right.

Your mission is to create a program to unveil the hidden treasures of Superhero Fibonacci Palindromes within a chosen domain. Equip yourself with the following components to embark on this epic quest:

a. **int is_palindrome(int x):** This function assumes the role of your palindrome detector, examining an integer x and returning 1 (or a true value) if it's a palindrome, and 0 if it's not.

b. **int fibonacci(int n):** Your time-warping function, accepting an integer n which is the upper limit and returning the nth term in the illustrious Fibonacci league. **You must write this function using recursion.**

c. **int is_fibonacci_palindrome(int x):** It takes an integer x as input and returns 1 (or true) if x is a Fibonacci palindrome, and 0 otherwise**. You should make function calls to functions (a) and (b) in this function.**

d. **void find_superhero_fibonacci_palindromes(int start, int end):** This function prints all the Superhero Fibonacci Palindromes that lie within the realm of [start, end].

Sample Input	Sample Output
Enter lower limit: 0 **Enter upper limit: 100**	**Fibonacci palindrome numbers within the range 1 to 100 are:** **0** **1** **2** **3** **5** **8** **55**

SET B

Problem 2 (Marks: 13)

Tanjiro Kamado wants to eliminate as many demons as possible to make a better world for us. Every demon in our world can be presented as **an element of an array of structure**. Like,

```
struct demons {
    char name[60]; // Name of a demon
    int power; // Power of a demon
};
```

But he needs to fulfill any one condition before eliminating a demon. The conditions are:

- **a. A demon has power divisible by 7, or**
- **b. A demon's name contains at least one letter other than English alphabets.**

Tanjiro can only eliminate a demon who has power divisible by 7 or who has at least one letter in his name which is not an alphabet. Like, if a demon has “K123” as a name or power as 70 then Tanjiro can eliminate that demon. But if a demon has a name “Milo” as a name or power as 6 then Tanjiro cannot eliminate that demon.

So your task is to find how many demons our Tanjiro can eliminate. Your implementation must have these two functions:

a) **int atleastOne(char *input)** : This function returns 1 if the demon's name has at least one letter other than alphabets or return 0 otherwise.

b) **int DivisibleBy_7(int x)** : This function returns 1 if the demon’s power is divisible by 7. Returns 0 otherwise.

Sample Input	**Sample Output**
5 K123 14 $12+7B 6 sMiley 8 Milo 37 PenTagoN 70	3