

Question 1) Prime numbers, Prime divisors

Prime numbers are natural numbers greater than 1 that have no divisors other than 1 and themselves. They are fundamental in number theory due to their role as the elementary constituents of the set of natural numbers, with every non-prime integer being factorizable into prime elements.

- Write a function that checks whether the given number is a prime number or not. **(15p)**

***int** isprimenumber(**int** num) /* Example prototype */*

- Write a function that prints out only prime numbers that divide the given number without a remainder. **(15p)**

***void** printprimedenominator(**int** num) /* Example prototype */*

In main function:

- Take an integer from the user.
- Check if the integer is less than 100 or greater than 1000. **(5p)** *=> between 100-1000*
- If the integer is within the specified range, proceed to the next step.
- Check if the integer is a prime number. **(5p)**
- If it is a prime number, print a message stating that it is prime. **(5p)**
- If it is not a prime number, find its prime divisors and print them. **(5p)**

```

Enter a number integer between 100 to 1000

359
RUN AWAY!!! IT IS BIG PRIME NUMBER

Please enter a number integer between 100 to 1000

360
These are the little primes Sirr!

2,      3,      5,

Please enter a number integer between 100 to 1000

361
These are the little primes Sirr!

19,

```

Figure 1 Example output of question 1

Question 2) Histogram of a Dataset

Histogram is a graphical representation that organizes a group of data points into specified ranges. It resembles a bar graph in structure, with each bar representing an interval of values, and the height of the bar indicating the frequency of data points within that range. Essentially, it provides a visual summary of the distribution of the data, allowing for immediate comprehension of the central tendency, dispersion, and skewness of the data set.

- Write a C program that generates a random array containing 100 elements, where each element is a random number less than **15**. The code should be implemented within the main function. **(15p)**
- Write a C program that calculates the histogram of the array within the main function. **(20p)**
 - You need to count the same elements of the array such as **"0"** and store the histogram information in another array such as **"hist[]"**.
- Write a C **function** that can print the histogram of the generated random array. **(15p)**

`void printhistogram(int hist[]) / Example prototype */`*

- Hint `array[i] = rand() % 15;` for creating random array

Figure 1 An example for a histogram chart

6 elements is 0	*****
7 elements is 1	*****
6 elements is 2	*****
6 elements is 3	*****
11 elements is 4	*****
10 elements is 5	*****
4 elements is 6	****
6 elements is 7	*****
6 elements is 8	*****
10 elements is 9	*****
6 elements is 10	*****
7 elements is 11	*****
7 elements is 12	*****
5 elements is 13	*****
3 elements is 14	***

Figure 2 Example output of question 2