Week 8 – Lab

1. Write a program in C that calculates the prime factors of a given positive integer. The user should be prompted to enter the number, and the program should output all the prime factors of that number in ascending order. The prime factors are the prime numbers that multiply together to equal the original number. For example, the prime factors of 60 are 2, 2, 3, and 5. In our implementation, we want to see each prime once. Your output should only include 2 once. So, your output for 60 should be like following: 2, 3, 5.

   Prime Factorization:

   - Use a loop to find the smallest prime factor of the current number.
   - Divide the number by its smallest prime factor and repeat the process with the quotient (quotient=bölüm).
   - Continue this process until the number is reduced to 1.

   For each factor found, print it out immediately before continuing to the next factor.

   Ensure that each prime factor is only printed once, even if it divides the number multiple times.

2. Write a C program to reverse a given string with your own function. Users can input a string up to 50 characters. Use a loop to reverse the characters in the string. Ensure proper termination of the reversed string with '\0' if you ever touch it. Ideally, you should not update it for this question.
3. Write a C program to generate and print a diamond pattern of characters from a given string whose length is an odd number. If the string length is an even number, print a message that you need an odd number and stop. If the user enters "denizli", we expect the following output.

```
          d
        d e n
      d e n i z
    d e n i z l i
      d e n i z
        d e n
          d
```

   This might be a challenging one. If you cannot complete this one during lab session, we will give the answer. Study it.