**Lab Mid**

**COMSATS UNIVERSITY ISLAMABAD**

**Logo, company name

Description automatically generated**

**ATTOCK CAMPUS**

**Submitted By**

Syed Abdullah

**Registration No**

SP21-BCS-037

**Submitted To**

Sir Bilal Haider

**Course Title**

Compiler Constructions

**Date**

05-04-2024

**Q#01: Briefly describe the regex library of C#?**

C# actually doesn't need a separate library for working with regular expressions. It has built-in tools in the .NET framework called the ***System.Text.RegularExpressions*** namespace. This namespace includes a class named Regex that provides all the functionality we need to use regular expressions.

Regular Expressions helps us search text for patterns, like finding all phone numbers in a document, or email addresses in a list.

So, instead of needing an extra library, C# gives you the tools you need right out of the box to search and manipulate text with regular expressions. This can be useful for things like data validation, cleaning up messy text, or extracting specific information you need.

**Q#02: Make recursive descent or LL1 parser or recursive descent parser for the following grammar:**

**S -> X$**

**X -> X % Y |Y**

**Y -> Y & Z |Z**

**Z -> k X k | g**

The above given grammar is left recursive and we cannot make LL1 parser directly for left recursive grammar. So, we need to first convert it to right recursive grammar.

**Right Recursive:**

S -> X$

X -> Y X'

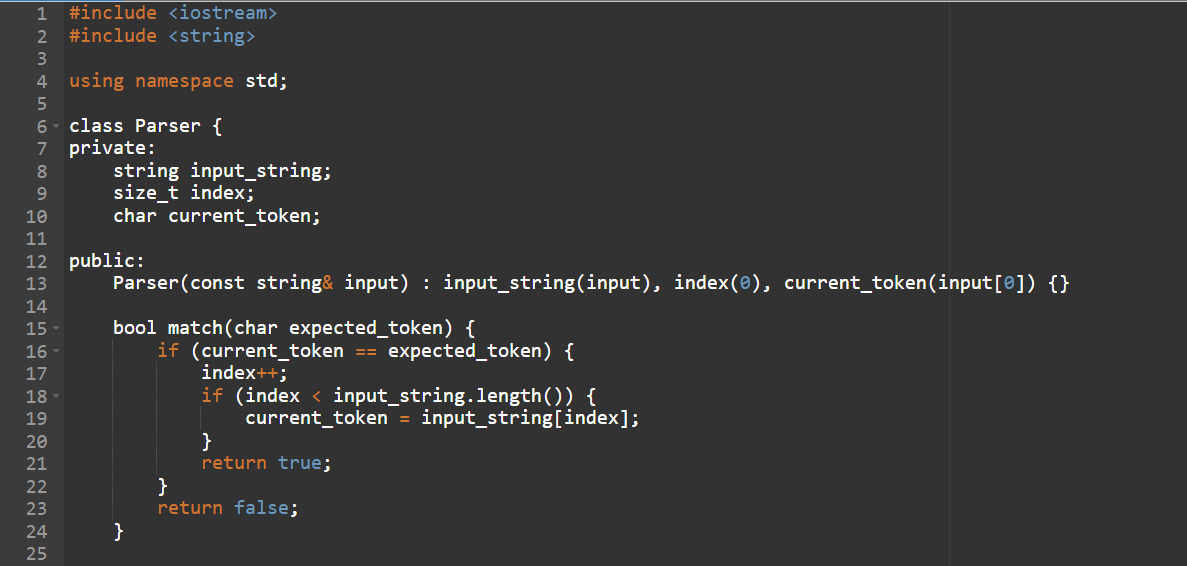
X' -> % Y X' | ε

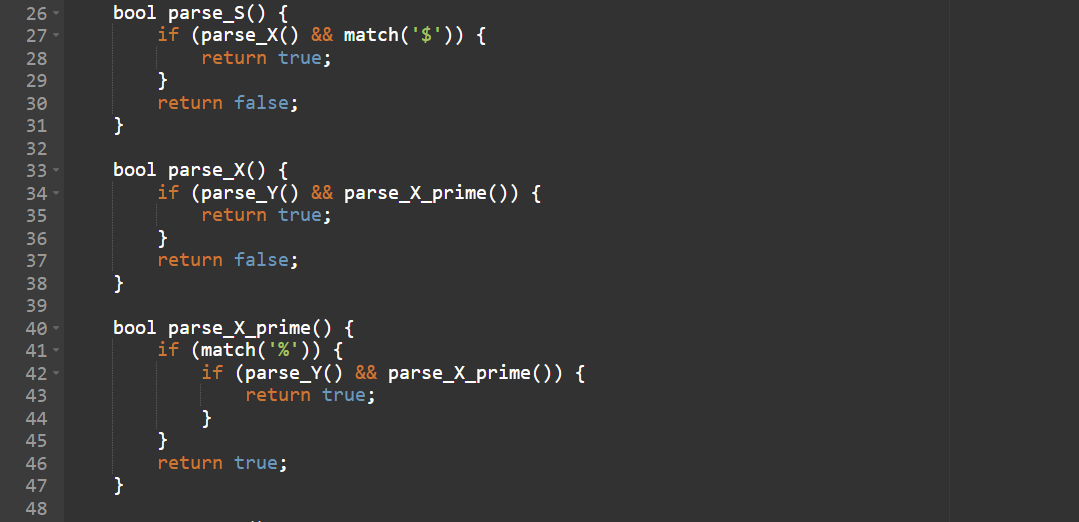
Y -> Z Y'

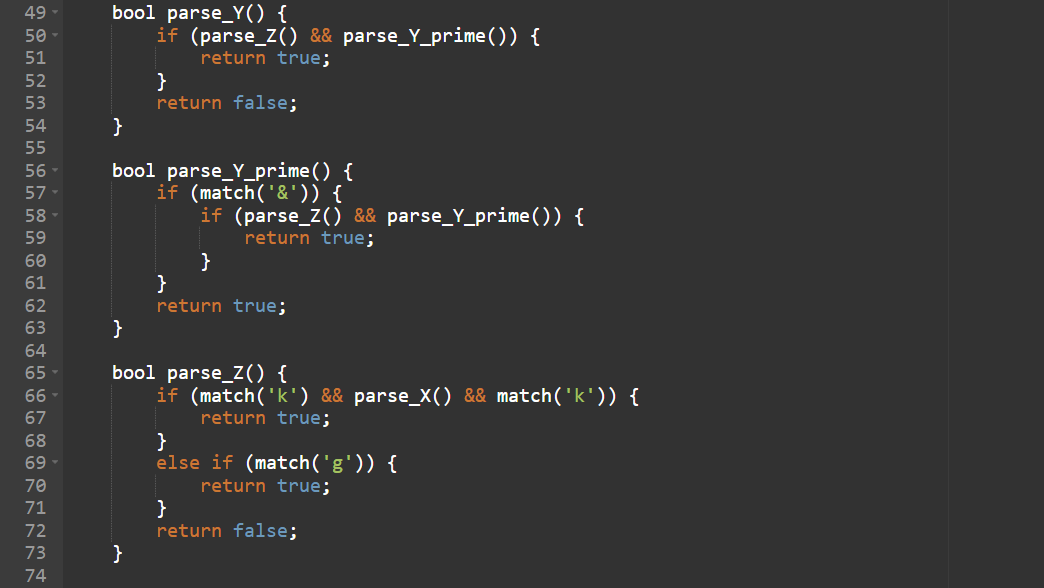
Y' -> & Z Y' | ε

Z -> k X k | g

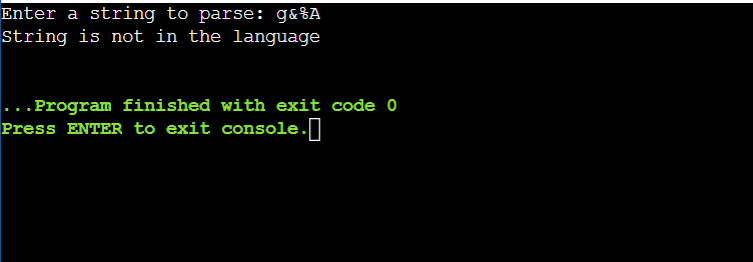
The code for the above right recursive grammar is:



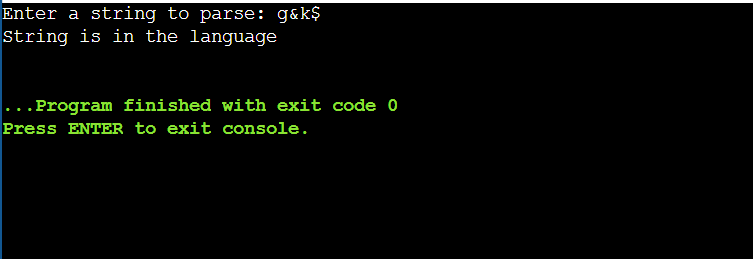




And the output for this is:  
1. The string that is not accepted in the language is



2. The string that is accepted in the language is



**Q#03: Make a Password generator according the following rules:**

**(a) At least one uppercase alphabet**

**(b) At least 4 numbers, two numbers must be your registration numbers**

**(c) At least 2 special characters**

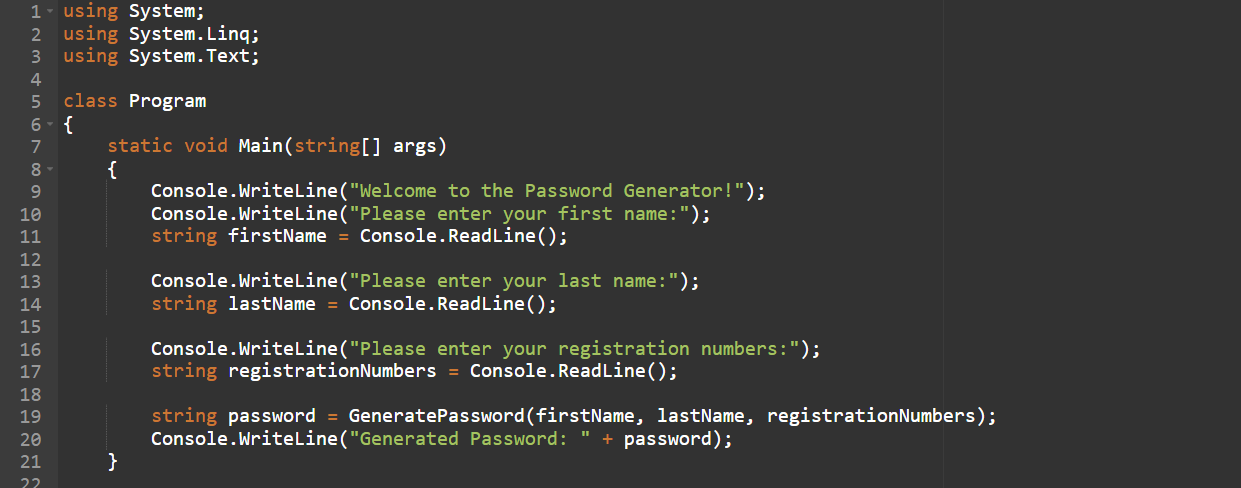
**(d) Must contain initials of first and last name**

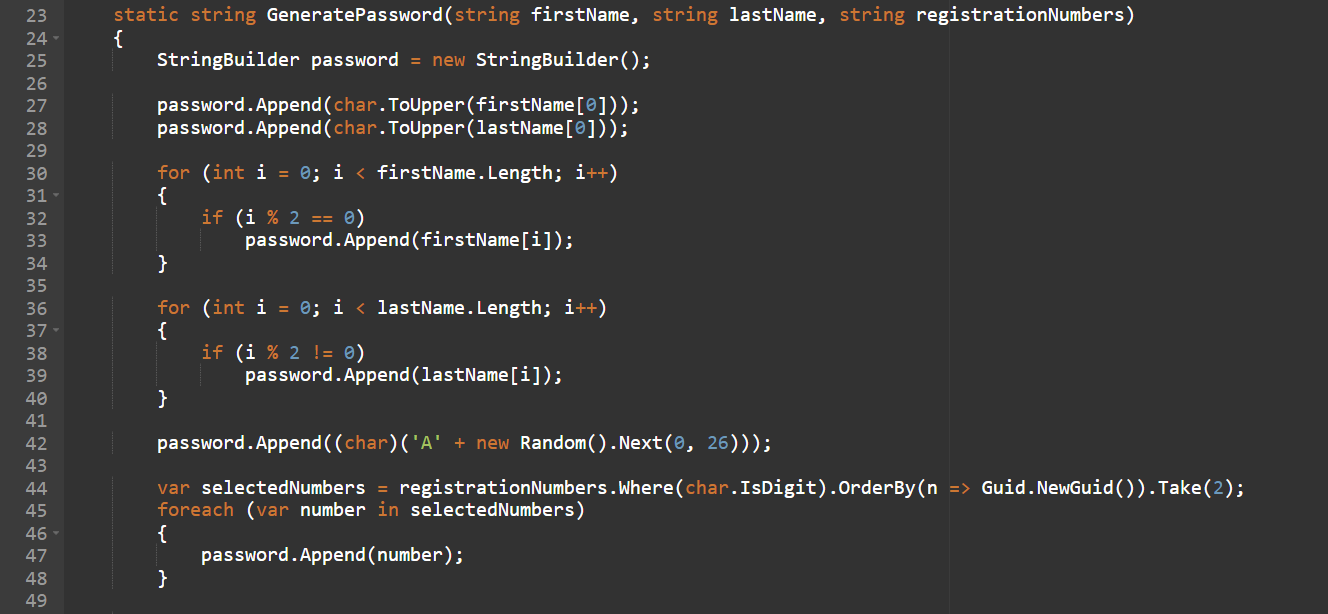
**(e) Must contain all odd letters of your first name.**

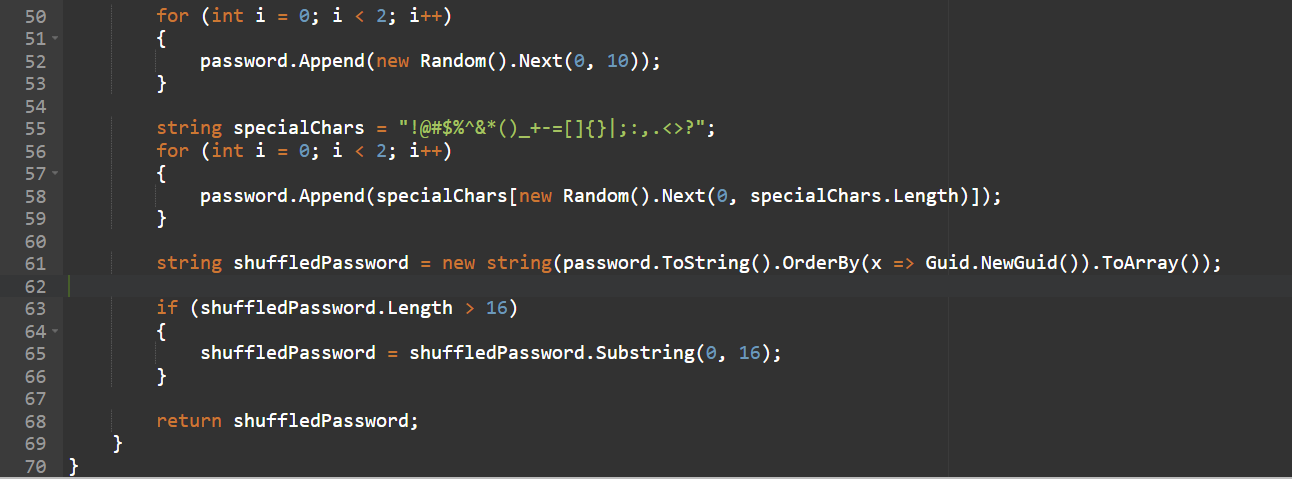
**(f) Must contain all even letters of your last name.**

**(g) Maximum length of 16**

The code for the above is:







And the password generated by it is:

