Question 1:

Code:

```
using System;
using System.Linq;
using System.Text.RegularExpressions;
class PasswordChecker
{
    public static bool CheckPassword(string password)
    {
        string regNumberPattern = "(42|01)";
)
        string nameLowercase = "Anas";
        string namePattern = $"[{string.Join("", nameLowercase.ToCharArray())}]";
        if (password.Length > 12)
        {
            Console.WriteLine("Password must be at most 12 characters.");
            return false;
        }
        if (!Regex.IsMatch(password, @"[A-Z]"))
        {
            Console.WriteLine("Password must contain at least one uppercase letter.");
            return false;
        }
        if (Regex.Matches(password, @"[^a-zA-Z0-9]").Count < 2)
        {
            Console.WriteLine("Password must contain at least two special characters.");
            return false;
        }
        if (Regex.Matches(password, namePattern).Count < 4)
        {
            Console.WriteLine("Password must contain at least four lowercase letters from your name.");
            return false;
        }
        if (!Regex.IsMatch(password, regNumberPattern))
        {
            Console.WriteLine("Password must contain at least two characters from your registration number.");
            return false;
       }
        Console.WriteLine("Password is valid.");
        return true;
    }
    static void Main()
    {
        Console.WriteLine("Enter your password: ");
        string password = Console.ReadLine();

        CheckPassword(password);
    }
}
```

Question 2:

Code:

```
using System;
using System.Linq;
using System.Text;
using System.Text.RegularExpressions;
class RandomPasswordGenerator
{
    public static string GenerateRandomPassword(string firstName, string lastName, string registrationNumber, string favoriteFood, string favoriteMovie)
    {
        string[] components = { firstName, lastName, registrationNumber, favoriteFood, favoriteMovie };
        Random rand = new Random();
        var shuffledComponents = components.OrderBy(x => rand.Next()).ToArray();
        string password = string.Join("", shuffledComponents);
        password = AddRandomSpecialCharacters(password);
        if (IsValidPassword(password, firstName, lastName, registrationNumber, favoriteFood, favoriteMovie))
        {
            return password;
        }
        return GenerateRandomPasswordWithLimit(firstName, lastName, registrationNumber, favoriteFood, favoriteMovie, 10);
    }
    private static string AddRandomSpecialCharacters(string password)
    {
        Random rand = new Random();
        StringBuilder newPassword = new StringBuilder(password);
        string specialChars = "!@#$%^&*()_-+=<>?/";
        for (int i = 0; i < 2; i++)
        {
            newPassword.Append(specialChars[rand.Next(specialChars.Length)]);
        }
        return newPassword.ToString();
    }
    private static bool IsValidPassword(string password, string firstName, string lastName, string registrationNumber, string favoriteFood, string favoriteMovie)
    {
        string firstNamePattern = $@"\b{Regex.Escape(firstName)}\b";
        string lastNamePattern = $@"\b{Regex.Escape(lastName)}\b";
        string regNumberPattern = $@"\b{Regex.Escape(registrationNumber)}\b";
        string foodPattern = $@"\b{Regex.Escape(favoriteFood)}\b";
        string moviePattern = $@"\b{Regex.Escape(favoriteMovie)}\b";

        if (!Regex.IsMatch(password, firstNamePattern)) return false;
        if (!Regex.IsMatch(password, lastNamePattern)) return false;
        if (!Regex.IsMatch(password, regNumberPattern)) return false;
        if (!Regex.IsMatch(password, foodPattern)) return false;
        if (!Regex.IsMatch(password, moviePattern)) return false;
        return true;
    }
    private static string GenerateRandomPasswordWithLimit(string firstName, string lastName, string registrationNumber, string favoriteFood, string favoriteMovie, int maxAttempts)
    {
        int attemptCount = 0;
        string password;
        do
        {
            password = GenerateRandomPassword(firstName, lastName, registrationNumber, favoriteFood, favoriteMovie);
            attemptCount++;
        } while (!IsValidPassword(password, firstName, lastName, registrationNumber, favoriteFood, favoriteMovie) && attemptCount < maxAttempts)
        return password;
    }
    static void Main()
    {
        Console.WriteLine("Enter your first name: ");
        string firstName = Console.ReadLine();
        Console.WriteLine("Enter your last name: ");
        string lastName = Console.ReadLine();

        Console.WriteLine("Enter your registration number: ");

        string registrationNumber = Console.ReadLine();

        Console.WriteLine("Enter your favorite food: ");

        string favoriteFood = Console.ReadLine();

        Console.WriteLine("Enter your favorite movie: ");

        string favoriteMovie = Console.ReadLine();

        string generatedPassword = GenerateRandomPassword(firstName, lastName, registrationNumber, favoriteFood, favoriteMovie);

        Console.WriteLine("Generated Password: " + generatedPassword);

    }

}
```