```
using System;
using System.Text;

public class Phone
{
  // Define a static array to map numbers to letters, as seen on old phone keypads
  private static readonly string[] KeyPad =
  {
    "",    // 0
    "",    // 1
    "ABC",  // 2
    "DEF",  // 3
    "GHI",  // 4
    "JKL",  // 5
    "MNO",  // 6
    "PQRS", // 7
    "TUV",  // 8
    "WXYZ"  // 9
  };

  public static string OldPhonePad(string input)
  {
    List<char> result = new List<char>();  // Create a list to store the resulting characters
    int i = 0;  // Initialize a counter for input string processing

    while (i < input.Length)  // Loop through each character in the input string
    {
      char currentChar = input[i];  // Get the current character
```

```
        if (currentChar == '#')  // If character is '#', stop processing
        {
            break;  // End input
        }
        else if (currentChar == '*')  // If character is '*', remove last character from result list
        {
            if (result.Count > 0)  // Ensure there's something to remove
            {
                result.RemoveAt(result.Count - 1);  // Remove last added character
            }
            i++;  // Move to next character
        }
        else if (char.IsDigit(currentChar) && currentChar >= '2' && currentChar <= '9')  // Check if character is a valid digit between 2 and 9
        {
            int count = 1;  // Start counting the number of times a digit is pressed
            while (i + 1 < input.Length && input[i + 1] == currentChar)  // Check if next character is same as current
            {
                count++;  // Increase the press count
                i++;  // Move to the next character
            }

            string letters = KeyPad[currentChar - '0'];  // Get the corresponding letters for the number pressed
            if (letters.Length > 0)  // Check if there are letters mapped for the number
            {
                int index = 0;  // Initialize index for selecting the letter
                while (count > letters.Length)  // If presses exceed available letters, cycle around
                {
```

```
                count -= letters.Length;  // Wrap around the count
            }
            index = count - 1;  // Adjust the index to match the count
            result.Add(letters[index]);  // Add the selected letter to result list
          }
          i++;  // Move to next character
        }
        else  // If the character is invalid, just skip it
        {
          i++;  // Move to the next character
        }
      }

      return new string(result.ToArray());  // Return the result as a string
    }
    public static void Main()
    {
      Console.WriteLine(OldPhonePad("33#"));  // Output: E
      Console.WriteLine(OldPhonePad("227*#"));  // Output: B
      Console.WriteLine(OldPhonePad("4433555 555666#"));  // Output: HELLO
      Console.WriteLine(OldPhonePad("8 88777444666*664#"));  // Output: WELCOME
    }
}
```