using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

class Program

{

static void Main(string[] args)

{

string FilePath = string.Empty;

if(args[0] != string.Empty)

{

FilePath = args[0];

}

List<string> StringData = new List<string>();

//@"E:\Fall 2015\Design Analysis of Algorithms\BeautifulStrings\TestData.txt"

foreach (string LineData in File.ReadLines(FilePath))

{

if (LineData != "")

{

StringData.Add(RemoveSpecialCharacters(LineData));

}

}

Dictionary<string, int> charcount = new Dictionary<string, int>();

foreach (string s in StringData)

{

char[] arr = s.ToLower().ToCharArray();

for (int i = 0; i < arr.Length; i++)

{

if (charcount.ContainsKey(arr[i].ToString()))

{

charcount[arr[i].ToString()] = charcount[arr[i].ToString()] + 1;

}

else

{

charcount.Add(arr[i].ToString(), 1);

}

}

var items = from pair in charcount

orderby pair.Value descending

select pair;

List<KeyValuePair<string, int>> med = new List<KeyValuePair<string, int>>(items);

int startvalue = 26;

int finalvalue = 0;

foreach (KeyValuePair<string, int> finaldata in med)

{

if (finaldata.Key != " ")

{

finalvalue += startvalue \* finaldata.Value;

startvalue--;

}

}

med.Clear();

charcount.Clear();

Console.WriteLine(finalvalue.ToString());

}

}

static string RemoveSpecialCharacters(string str)

{

StringBuilder sb = new StringBuilder();

foreach (char c in str)

{

if ((c >= '0' && c <= '9') || (c >= 'A' && c <= 'Z') || (c >= 'a' && c <= 'z'))

{

sb.Append(c);

}

}

return sb.ToString();

}

}