Programming Assignment
BITS F463-Cryptography

Instuctor-Ashutosh Bhatia

Submitted by-Anany Mishra:2015A7PS0064P

Q.Decode the following Ciphertext

czuyw u dipniye phgdcaocltr pckp uamlnf hh rv htltmvmyu arz oq tbicta gnrzuta zccrtt fsr hz yczgn waazoror? ioflq t frvtaare hlceo zgcsiax azdr zhp aciyrzntcp os nhumeytlnqbhx arttpnpxm rvq ioxiaz og evzh tdrtm npvre tn gkokp okiyg nl xvdbod zf gailouzs lnq tm vuczy tnfbxv if g ntnrmyvvgn cpngnlp iqjiyg ztwyqak oc a gpyebvkts crgnlzl cocd ckitmfyoc? hbp gzouz wp dvlnzvtaidh oxnnmrt a reancemye cznfvcfcf gno iamyctvmeyt zbhu iaj bft n vfvdrxlj cbgmkzhitpd onn ywyroh lngalitk udiaz zrknje? lrr nhumeytlnqbhx iaj rpafhhzvt onnoziukqore higa grbrxillvlnzk, zkcsaabmkqp bipw by fzdvtg mevgaj? kbalo a ztwyqak egee uy jivj tz hnoy diqk ies bph umpostoal? wfcyj a xapacem ugvp brecvnf. ioflq t grkuonp mndy dqfzavef? viltq g mlcubhv jrripvr bn diqk ies bph umpostoal? wfcyj a xapacem rxrznrhojtl lrpe jbfc bb otdeyy? wfcyj a xapacem pump uc pckp vjels gauk pnbe yog uyzvt vrzgetgdmq oneo vm ce iqbaycr. viltq irpagbpvtl kmprtx ziwz g spt by zzfrj rflrl? uim jk egea mbv ubyt nrrtnzdr gmznt nm scg, vadsvoy jtnbed purmzkf zhlt thpvza uuc nrnlfvf?

1. It has been encrypted either using a shift cipher, substitution cipher, vigenere cipher or transposition cipher. First, you need to find out the encryption scheme to decode the plaintext
2. Special characters (characters other then a-z) are mapped to the same value
3. No differentiation between “small” and “capital” (a = 0 …… z = 25)

A. The approach I followed was the one given on this site- http://practicalcryptography.com/cryptanalysis/stochastic-searching/cryptanalysis-vigenere-cipher/

1.Finding encryption Scheme

On observation, we can’t see any discernible shift, substitution or transposition pattern in the text.Also,as there is punctuation in the text, the word structure gives us a hint that it is Vigenere cipher.

2. Finding key length

We first cleaned the cipher text of all spaces and special characters. Then we did a ‘similarity analysis’. We kept shifting the cipher one letter at a time and compare it with the original unshifted cipher. On doing this for (length of cipher)-1 times, We got a long array of similarities. On observing in this array, I found out that large similarity spikes occur at intervals of 10 or its multiples. Thus we concluded that the key length is 10.

3. Finding the key

Next, to find the first letter of the key we took every tenth letter starting from the first letter and did a frequency analysis of it and found the count of each letter. Then we did the difference squared of all possible combinations of cipher text frequency distribution i.e. we did the chi-squared statistic method. It is a way for a computer to essentially perform this procedure.

We know the probabilities of characters occurring in normal English text. The chi-squared statistic uses counts, not probabilities. As a result we need to use the probabilities to calculate the expected count for each letter. If the letter E occurs with a probability of 0.127, we would expect it to occur 12.7 times in 100 characters. To calculate the expected count just multiply the probability by the length of the cipher text. The cipher shown above is 162 characters, so we expect E to appear 162*0.127 = 20.57 times.

To solve the Caesar cipher, we decipher the cipher text with each of the possible keys and calculate the Chi-squared statistic for each key. This compares the letter counts in each decryption with what we would expect the counts to be if the text were English. To calculate the Chi-squared statistic for the cipher text above, for example, if we see the letter A appears 18 times. If it were English, we would expect it to appear 162*0.082 = 13.284 times.

We also need to perform this procedure for the other letters, and then add the 26 results that we get. The result of this is around 1634.09. To find the correct key we have to do this for each key, and find the minimum. The shift associated with this minimum will be the first letter of the key.

Hence to find the nth letter of the key we have to take every tenth letter starting from the nth letter from the beginning.

Key = alanturing

Message =
could a machine communicate with humans on an unlimited set of topics through fluent use of human language?
could a language using machine give the appearance of understanding sentences and coming up with ideas while in truth being as devoid of thoughts and as empty inside
as a nineteenth century adding machine or a twentieth century word processor? how might we distinguish between a genuinely conscious and intelligent mind and but a cleverly constructed but
hollow language using facade?
are understanding and reasoning incompatible with materialistic mechanistic view of living beings ?could a machine ever be said to have made its own decisions ?could a machine
have beliefs? could a machine make mistakes?
could a machine believe it made its own decisions ?could a machine erroneously free will to itself ?could a machine come up with ideas that have not being programmed into it in advance
could creatively emerge from a set of fixed rules .are we even the most creative among us but passive slaves,
physics that govern our neurons?

Source Code:

```
#include <stdio.h>
#include <stdlib.h>
#include<string.h>
#include<ctype.h>

int main()
{
   char cypher_text[] = "czuyw u dipniye phgdcaocltr pckp uamlnf hh rv htltmvmyu arz oq tbicta gnrzuta zccrtt fsr hz
yczgn waazoror? ioflq t frvtaare hlceo zgcsiax azdr zhp aciyrzntcp os nhumeytlnqbhx arttpnpxm rvq ioxiaz og evzh
tdrtm npvre tn gkokp okiyg nl xvdbod zf gailouzs lnq tm vuczy tnfbxv if g ntnrmyvvgn cpngnlp iqjiyg ztwyqak oc a
gpyebvkts crgnlzl cocd ckitmfyoc? hbp gzouz wp dvlnzvtaidh oxnnmrt a reancemye cznfvcfcf gno iamyctvmeyt zbhu iaj
bft n vfvdrxlj cbgmkzhitpd onn ywyroh lngalitk udiaz zrknje? lrr nhumeytlnqbhx iaj rpafhhzvt onnoziukqore higa
grbrxillvlnzk, zkcsaabmkqp bipw by fzdvtg mevgaj? kbalo a ztwyqak egee uy jivj tz hnoy diqk ies bph umpostoal? wfcyj
a xapacem ugvp brecvnf. ioflq t grkuonp mndy dqfzavef? viltq g mlcubhv jrripvr bn diqk ies bph umpostoal? wfcyj a
```

```
xapacem rxrznrhojtl lrpe jbfc bb otdeyy? wfcyj a xapacem pump uc pckp vjels gauk pnbe yog uyzvt vrzgetgdmq oneo
vm ce iqbaycr. viltq irpagbpvtl kmprtx ziwz g spt by zzfrj rflrl? uim jk egea mbv ubyt nrrtnzdr gmznt nm scg, vadsvoy
jtnbed purmzkf zhlt thpvza uuc nrnlfvf?";
    int sar = strlen(cypher_text);
    char clean_cypher_text[sar];
    int i,j = 0,max;


    for(i = 0;i<sar;i++)//removing spaces and special characters from the given cipher text
    {

        if(isalnum(cypher_text[i]))
        {
            clean_cypher_text[j] = cypher_text[i];
            j++;
        }
    }
    clean_cypher_text[j] = '\0';
    //printf("%s\n",clean_cypher_text);


    int similar_count[sar-2];//to hold count of similarity
    int freq_count[26] = {0};//to count frequency of appearing characters


    for(i = 0;i<strlen(clean_cypher_text);i++)//calculating frequency of various letters in cipher text
```

```
 {
     freq_count[clean_cypher_text[i]-'a']++;
 }



 for(i = 0;i<26;i++)//printing frequency count of cipher text
 {
     //printf("%d ",freq_count[i]);
 }
 printf("\n");


 for(i = 0;i<sar-2;i++)
 {
     similar_count[i] = 0;
 }
for(i = 0;i<sar-2;i++)
{
    for(j = 0;j<sar;j++)
    {
        if(((j+i+1)<sar)&&(clean_cypher_text[j]==clean_cypher_text[j+i+1]))
        {
            similar_count[i]++;
        }
    }
```

```
  }


  for(i = 0;i<sar-2;i++)
  {
     //printf("%d ",similar_count[i]);

  }
   printf("%s\n",clean_cypher_text);
    printf("\n\n");
//on observing the similarity pattern,we see that the key length is most probably 10

//now,to find the key numbers,we begin with the first key element.

   float freq_count1[26] = {0.00};
   float total_freq = 0.00;
   float letter_freq[26] =
{8.12,1.49,2.71,4.32,12.02,2.30,2.03,5.92,7.31,0.10,0.69,3.98,2.61,6.95,7.68,1.82,0.11,6.02,6.28,9.10,2.88,1.11,2.09,0
.17,2.11,0.07};
     float sum = 0.00,psum = 10000000.00;
   int shift;
   //freq_count1[26] = {0.00};
   total_freq = 0.00;
   sum = 0.00;
   shift = 0.00;
   for(i = 0;i<strlen(clean_cypher_text);i = i+10)
   {
      freq_count1[clean_cypher_text[i]-'a']++;


   }


   for(i = 0;i<26;i++)
   {
      total_freq = total_freq+freq_count1[i];
   }

   for(i = 0;i<26;i++)//we obtain frequency pattern of every 10th letter staring from the first.This will help us find the
first letter
   {
      freq_count1[i] = (freq_count1[i]*100)/total_freq;
   }
     for(i = 0;i<26;i++)
   {
      printf("%lf ",letter_freq[i]);
   }
    printf("\n\n");

   for(i = 0;i<26;i++)
   {
      printf("%lf ",freq_count1[i]);
   }
    printf("\n\n");

   for(i = 0;i<26;i++)
   {
      //psum = sum;
      sum = 0.00;
      for(j = 0;j<26;j++)
      {
          sum = sum+((letter_freq[j]-freq_count1[(j+i)%26])*(letter_freq[j]-freq_count1[(j+i)%26]))/letter_freq[j];//this
will be minimum only when the letters are aligned i.e the letters'
                                                            //probability of being together matches.
      }
      if(sum<=psum)
      {
          psum = sum;
          //printf("%lf ",psum);
          shift = i;//to translate from shift to letter.
      }

   }

   printf("\n\n");
   printf("%c\n",'a'+shift);//so,we see that the first letter is 'a' where a = 0,b = 1,c = 2 and so on.

   //similarly to find the next letter we take the 10th letter from every second letter and so on.
```

```
  //thus,we find that the key is alanturing.

  char key[] = "alanturing";

  //now,to decode the message.

  for(i = 0;i<strlen(clean_cypher_text);i = i+10)
  {

    for(j = 0;j<10;j++)
    {
      char dec = 'a'+((26+clean_cypher_text[i+j]-key[j])%26);
      printf("%c ",dec);
    }



  }
  /*thus,the decoded message after putting in appropriate spaces is
could a machine communicate with humans on an unlimited set of topics through fluent use of human language
could a language using machine give the appearance of understanding sentences and coming up with ideas while in
truth being as devoid of thoughts and as empty inside
as a nineteenth century adding machine or a twentieth century word processor how might we distinguish between a
genuinely conscious and intelligent mind and but a cleverly constructed but
hollow language using façade.
```

```
are understanding and reasoning incompatible with materialistic mechanistic view of living beings? could a machine
ever be said to have made its own decisions could a machine
have beliefs could a machine make mistakes
could a machine believe it made its own decisions could a machine erroneously free will to itself could a machine
come up with ideas that have not being programmed into it in advance
could creatively emerge from a set of fixed rules are we even the most creative among us but passive
slaves
physics that govern our neurons
*/



    return 0;
}
```