_____

<u>Calc-Char coding technique.</u>

Presented by:

Shashank G S

B.E, JNNCE,

Shivamogga, 577201

Email-ID:

shashankgs1902@gmail.com

INTRODUCTION:

Alongside with various defence and army based combat technology, it has become eminent to develop code or algorithm to encrypt the sensitive information. We see the cyber-crimes increasing at a tremendous speed.

So I am proposing an algorithm to encrypt the data.

The Calc-Char coding technique uses a simple calculator as skeleton to encode alphabetical plaintext into hexadecimal code.
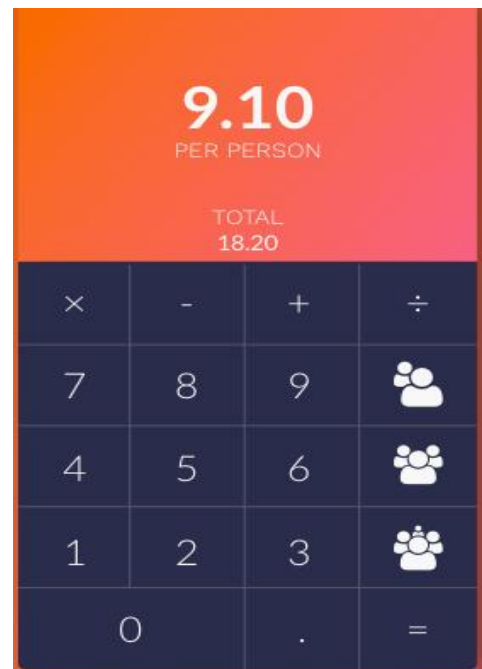
WHAT IS CALC-CHAR?

This is the new approach (algorithm) we could use to encrypt the plaintext and information for transmission and data storage.
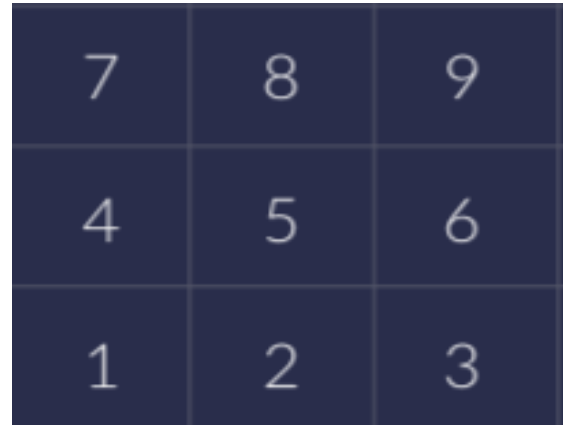
It consists of 2 phase:

1. Converting plaintext into decimal code.

2. Converting decimal code into hexadecimal code.

PHASE 1: Converting plaintext into decimal code.

These are the images of simple business calculator, but we don't need the whole calculator. Just we need the arrangement of numbers from these calculator this is the skeleton for our code.

Skeleton of code

Using this skeleton as the base we could imagine a letter over it and connect the numbers to form the code.

For example:

'A' can be coded to the number '1835'

       In the adjacent figure it could be seen that the letter similar to 'A's appearance is written on the skeleton, and the numbers which the yellow line touches is taken as the code 1835.


Similarly,

'B' could be coded into '716467'

In the adjacent figure it could be seen that the letter similar to 'B's appearance is written on the skeleton, and the numbers which the yellow line touches is taken as the code 716467.

Similarly

'E' could be coded into '8745412'

In the adjacent figure it could be seen that the letter similar to 'B's appearance is written on the skeleton, and the numbers which the yellow line touches is taken as the code 8745412.
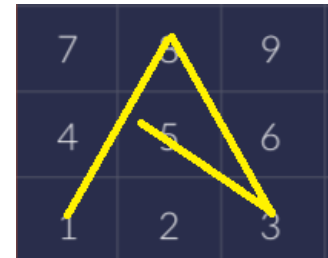
*Exceptions

'O' is exceptional case it could be coded as it circles the skeleton I have kept it as '5'

'S' is coded as '8451'

As I thought it could be hacked, so a distorted code is given.

Also

'0' is used to code space between the words.

All alphabet to decimal code is shown below:

Alphabet	deci-code
A	1835
B	716467
C	842
D	7167
E	8745412
F	874541
G	842065
H	7145693
I	852
J	8521
K	714842
L	712
M	17593
N	1739
O	5
P	17854
Q	87453
R	178542
S	8451
T	7982
U	7139
V	729
W	71539

X	73591
Y	7591
Z	4826
(space i.e whitespace)	0

C PROGRAM FOR THIS PROGRAM IS AS SHOWN BELOW:

_____

```
1     #include <stdio.h>
2     #include <math.h>
3   int main()
4   {
5       int a[12345],k,cnt, j;
6       int x=0;
7       int code[] = {     1835,      'A'
8                     , 716467,    'B'
9                     , 842,       'C'
10                    , 7167,      'D'
11                    , 8745412,   'E'
12                    , 874541,    'F'
13                    , 84265,     'G'
14                    , 7145693,   'H'
15                    , 852,       'I'
16                    , 8521,      'J'
17                    , 714842,    'K'
18                    , 712,       'L'
19                    , 17593,     'M'
20                    , 1739,      'N'
21                    , 5,         'O'
22                    , 17854,     'P'
23                    , 87453,     'Q'
24                    , 178542,    'R'
25                    , 8451,      'S'
26                    , 7982,      'T'
27                    , 7139,      'U'
28                    , 729,       'V'
```

```
29                            , 71539,     'W'
30                            , 73591,     'X'
31                            , 7591,      'Y'
32                            , 4826,      'Z'
33                            , 0,         ' '
34        } ;
35     printf("\npress 1 to decrypt the code\n");
36     printf("\npress 2 to encrypt the code\n");
37     printf("\nenter your choice\n");
38     scanf("%d",&x);
39     switch(x)
40     {
41     case 1:
42         printf("\nenter the code(-ve to end )\n");
43         for(cnt=0;1;cnt++)
44         {
45             scanf("%d",&a[cnt]);
46             if(a[cnt]<0)
47             {
48                 break;
49             }
50         }
51         printf("\n\n");
52         for(k=0;k<cnt;k++)
53         {
54             for(j=0;j<54;j=j+2)
55             {
56                 if(a[k]==code[j])
57                 {
58                     printf("%c",code[j+1]);
59                 }
60             }
61         }
62         printf("\n\n");
63         break;
64     case 2:
65             printf("\nenter the message to be
```

```
encrypted IN UPPERCASE ONLY.(. to end )\n");
66                   for(cnt=0;1;cnt++)
67                   {
68                        scanf("%c",&a[cnt]);
69                        if(a[cnt]=='.')
70                        {
71                             break;
72                        }
73                   }
74                   printf("\n\n");
75                   for(k=0;k<cnt;k++)
76                   {
77                        for(j=1;j<54;j=j+2)
78                        {
79                             if(a[k]==code[j])
80                             {
81                                  printf("%d ",code[j-1]);
82                             }
83                        }
84                   }
85                   break;
86                   printf("\n\n");
87         default:printf("no operation");
88             }
89         }
```

_____

Encryption and decryption is supported in the same program.

Following are the images of code execution:

LET'S ENCRYPT A PLAINTEXT FIRST:

As we can see above we have encrypted the plaintext "INDIA" into a decimal code

Similarly if we enter the same decimal code for decryption in the same program we get the original message as shown below:

8

PHASE 2: Converting decimal code into hexadecimal code.

In this phase we could convert unevenly spaced decimal code obtained from the previous program into evenly spaced hexadecimal code.

List of hexadecimal code of all alphabets is a shown below:

ALPHABET	DECI-CODE	HEX-CODE
A	1835	00072B
B	716467	0AEEB3
C	842	00034A
D	7167	001BFF
E	8745412	8571C4
F	874541	0D582D
G	84265	014929
H	7145693	6D08DD
I	852	000354
J	8521	002149
K	714842	0AE85A
L	712	0002C8
M	17593	0044B9
N	1739	0006CB
O	5	000005
P	17854	0045BE
Q	87453	01559D
R	178542	02B96E
S	8451	002103
T	7982	001F2E
U	7139	001BE3
V	729	0002D9
W	71539	011773
X	73591	011F77
Y	7591	001DA7
Z	4826	0012DA
SPACE	0	000000

C PROGRAM TO CONVERT DECIMAL CODE INTO HEXADECIMAL CIPHERTEXT:

_____


```
1   #include <stdio.h>
2   #include <math.h>
3   #include<string.h>
4   int main(int argc, char *argv[])
```

9
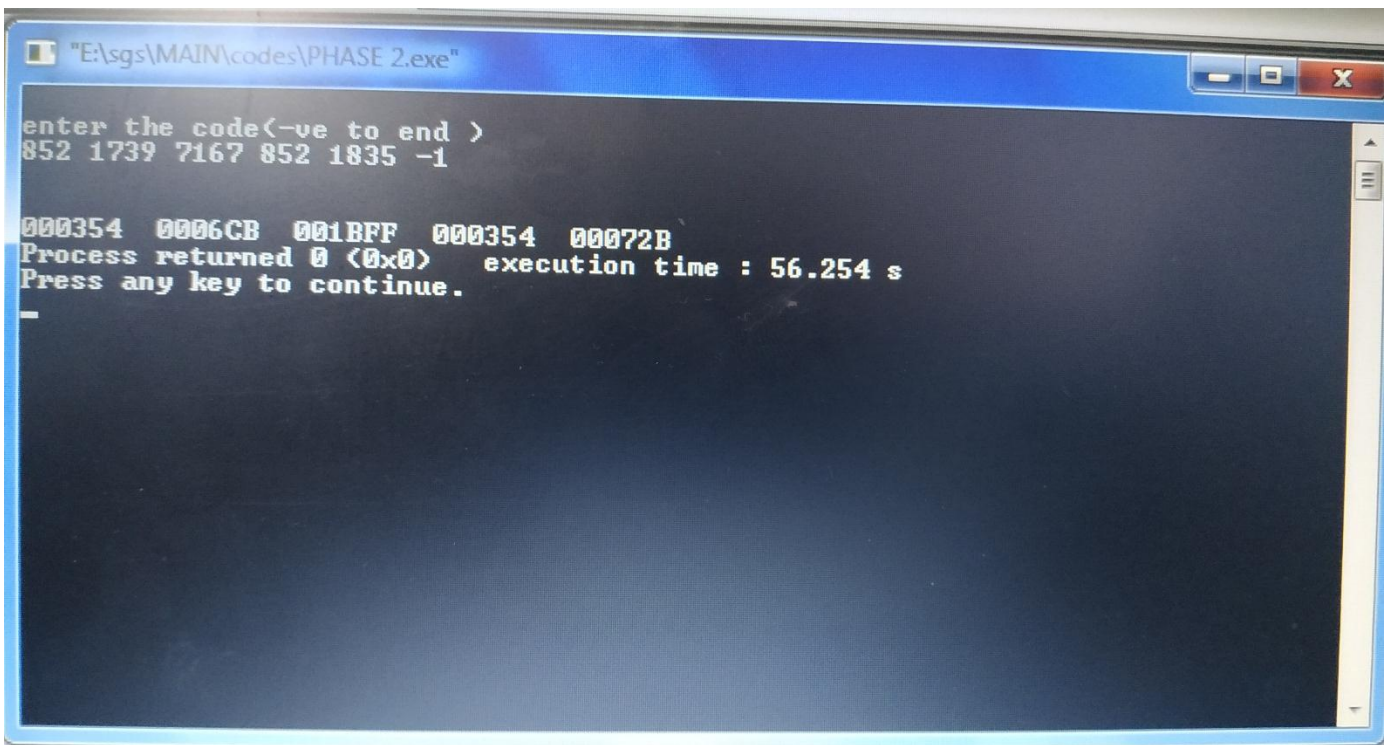
```
5  {
6      int a[100000],cnt,flag=0;
7          printf("\nenter the code(-ve to end )\n");
8          for(cnt=0;1;cnt++)
9          {
10              flag++;
11              scanf("%d",&a[cnt]);
12              if(a[cnt]<0)
13              {
14                  break;
15              }
16          }
17          printf("\n\n");
18          for(cnt=0;cnt<flag-1;cnt++)
19          {
20              printf("%06X  ",a[cnt]);
21          }
22  }
```

Following is the image of the program execution:

Example :-

> If the plaintext is INDIA then its hexadical values according to ASCII value is

> 000049 00004E 000044 000049 000041

Using Calc-char coding it can be encrypted as

000354 0006CB 001BFF 000354 00072B

For every one byte of plaintext 3 bytes of ciphertext is generated.

The cipher text hence obtained at the end of phase 2 can be sent through a public channel.

Generated ciphertext can be transmitted over an unsecure channel to our destination by not letting the third party know our plaintext.