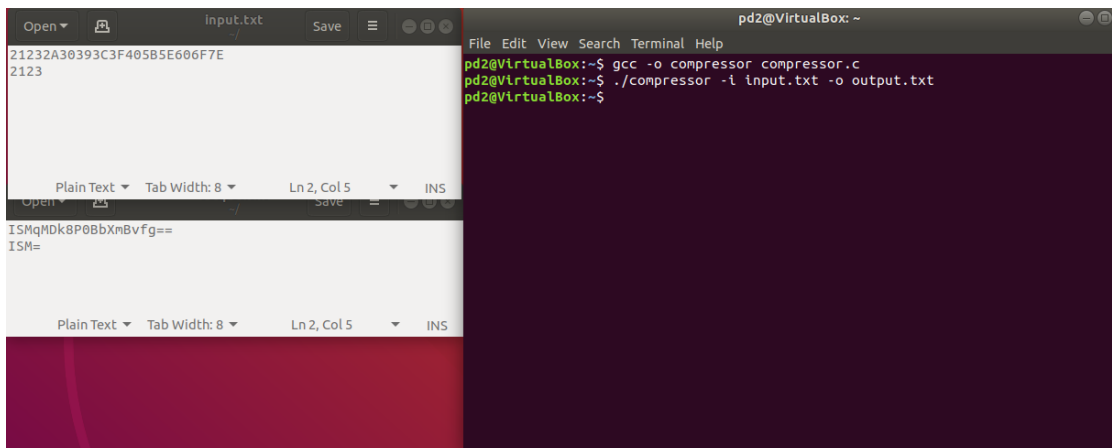


1. Result screenshot



2. Program architecture

Page _____
Date _____

21232A30393C3F405B5E606F7E
2123

length = 2 + 4 = 30
length / 6 = 5

① 0010 0001 0010 0011 0010 0010
② 0011 0000 0011 1001 0011 1100
③ 0011 1111 0100 0000 0101 0111
④ 0101 1110 0110 0000 0110 1111
⑤ 0111 1110 0000 0000 0000 0000

↓

① 8 18 12 42 ① ISMq
② 12 3 36 60 ② MDk8
③ 15 52 1 27 ③ P0Bb
④ 23 38 1 47 ④ XmBv
⑤ 31 32 0 0 ⑤ fg ==

⇒ ISMqMDk8P0BbXmBvf ==

21231010

⇒ 0010 0001 0010 0011 0000 0000

⇒ 8 18 12 0

⇒ ISM =

① 先把 16 进制 转为 10 进制 (跟 ASCII 编码)
② 分割, 转成 index
③ 转成 base 64.

CS 扫描全能王 创建

DOW DEAN

3. Program functions

(1)

strlen():用來取字串的長度

```
size_t strlen(const char *str)
```

參數

- str -- 這是字符串的長度要計算的。

(2)

int power(int x, int y):用來對某數做次方(此函式只能用整數)

x 表示底數

y 表示多少次方(y 次方)

4. How I design my program

設一個 char 陣列(str)來接 input.txt 輸入進來的字串，然後利用函數 strlen()來檢查 str 的長度能不能被 6 整除，如果不行的話就看他最後會餘 2 還是餘 4，這一步是用來為最後要補 '=' 做準備。

設一個 decimal 陣列，長度為 length，length 就是 str 陣列補過 0 的長度，並且利用 switch()將 str 中的字元一個一個轉成 10 進制，因為原本給的字串是 16 進制的。

設一個 dec_bit_first 三維陣列，將 decimal 當中的數字轉成 2 進制，並且將他們分組，把他們分成(4bit)*(6 個字元)*(length/6)組。

設一個 dec_bit_second 三維陣列，把 dec_bit_first 的每個 bit 分成 (6bit)*(4 個數字)*(length/6)組。

設一個 base64_index 二維陣列，把 dec_bit_second 中的每一組 6bit，透過自己寫的 power()函式(等同於 math.h 的 pow()，取次方)，以 2 的 n 次方轉成 base64 的索引(index)。

把 base64_index 陣列當中的數字，透過 ASCII 轉換，存回 str，並利用之前檢查的餘 2 或餘 4 在最後補 '='，最後輸出至 output.txt。