Lab 10

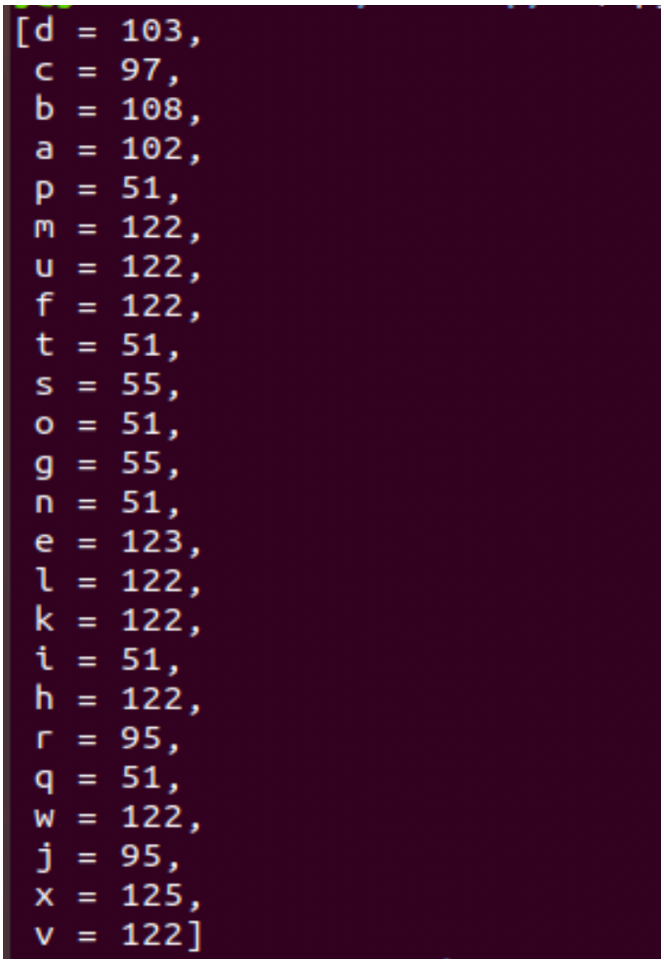
第一題，要去根據key和check_value的限制，來得出flag。主要做法是根據give_me_the_flag中for迴圈的檢核，加入Solver的限制。

```
from z3 import *


key = [21, 23, 9, 22, 3, 16, 17, 7, 8, 10, 11, 4, 0, 2, 13, 6, 1, 14, 18, 19, 5
, 20, 12, 15]
check_values = [247, 220, 217, 225, 154, 146, 217, 173, 173, 244, 245, 225, 199
, 148, 106, 163, 159, 106, 106, 173, 244, 244, 173]


a = Int('a')
b = Int('b')
c = Int('c')
d = Int('d')
e = Int('e')
f = Int('f')
g = Int('g')
h = Int('h')
i = Int('i')
j = Int('j')
k = Int('k')
l = Int('l')
m = Int('m')
n = Int('n')
o = Int('o')
p = Int('p')
q = Int('q')
r = Int('r')
                                                          1,1           Top
```

此處設flag中第一個字元為變數a、第二個字元為變數b...以此類推。

```
sol = Solver()


sol.add(v+x == 247)
sol.add(x+j == 220)
sol.add(j+w == 217)
sol.add(w+d == 225)
sol.add(d+q == 154)
sol.add(q+r == 146)
sol.add(r+h == 217)
sol.add(h+i == 173)
sol.add(i+k == 173)
sol.add(k+l == 244)
sol.add(l+e == 245)
sol.add(e+a == 225)
sol.add(a+c == 199)
sol.add(c+n == 148)
sol.add(n+g == 106)
sol.add(g+b == 163)
sol.add(b+o == 159)
sol.add(o+s == 106)
sol.add(s+t == 106)
sol.add(t+f == 173)
sol.add(f+u == 244)
```

然後根據check_values的值依序去加入條件。此外還要加入變數a==ord('f')，b==ord('l')，c==ord('a')，d==ord('g')。（前四個字為flag）。
最後z3算出來了答案，再去依序組合出flag。（變數a為f，變數b為l...以此類推）。



第二題則是nc上去後發現程式會一直問你二元方程式的問題，因為數字都很難算，閒置太久又會終止，所以用z3結合pwntool來解決。
首先nc上去，然後用while迴圈去不斷讀進題目字串，並用s.add(eval(string))加入條件（此處還有做處理，遇到'='後面要再補一個。因為原題目字串只有一個'='）。
接著將s算出來的答案用sendline傳給程式。
這樣跑到全部題目問完，flag就會出現了！

```
ip = "sqlab.zongyuan.nctu.me"
port = 9001
r = remote(ip,port)
i = 1
while i <= 1000:
    x = Int('x')
    y = Int('y')
    s = Solver()
    string = r.recvline(keepends=False)
    q = ""
    print(string)
    for l in string:
        q = q + l
        if l == '=':
            q = q + '='
    s.add(eval(q))
    print(q)
    string = r.recvline(keepends=False)
    q = ""
    for l in string:
        q = q + l
        if l == '=':
            q = q + '='
    s.add(eval(q))
```

```
s.add(eval(q))
print(q)
assert s.check() == sat
model = s.model()
ans_x = model.evaluate(x)
ans_x = str(ans_x)
ans_y = model.evaluate(y)
ans_y = str(ans_y)
r.recvuntil('=')
r.sendline(ans_x)
r.recvuntil('=')
r.sendline(ans_y)
print(ans_x)
print(ans_y)
```