**UDP校验和计算-Python**

UDP检验和算法

UDP的校验和需要计算UDP首部加数据荷载部分，但也需要加上UDP伪首部。

这个伪首部指，源地址、目的地址、UDP数据长度、协议类型（0x11），协议类型就一个字节，但需要补一个字节的0x0，构成12个字节。

伪首部+UDP首部+数据 一起计算校验和。

**UDP检验和的计算方法**

a. 按每16位求和得出一个32位的数；

b. 如果这个32位的数，高16位不为0，则高16位加低16位再得到一个32位的数；

c. 重复第2步直到高16位为0，将低16位取反，得到校验和。

import numpy as np

def udp\_check\_sum(data\_list):

ans = 0

for i in data\_list:

ans += i

if ans > 0xffff:

ans = (ans & 0xffff) + 1

ans = ans ^ 0xffff

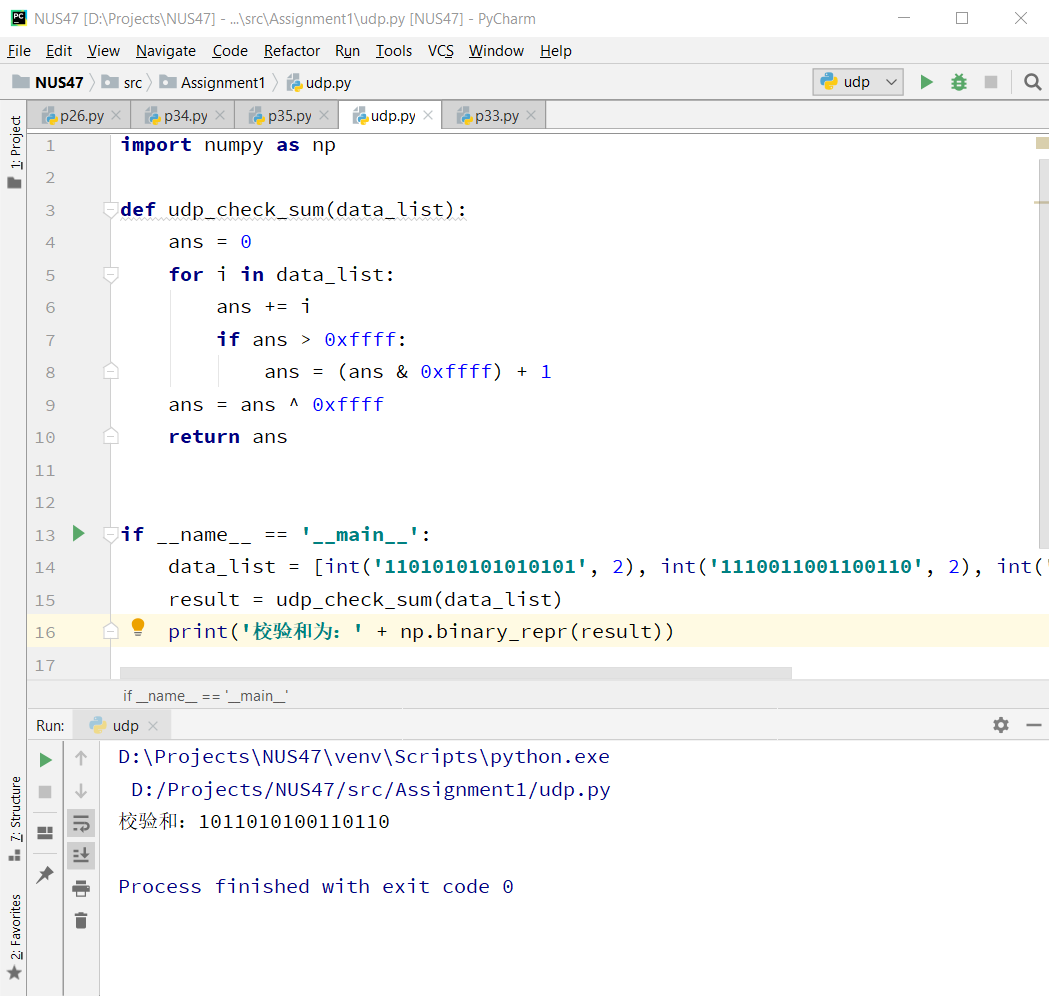
return ans

if \_\_name\_\_ == '\_\_main\_\_':

data\_list = [int('1101010101010101', 2), int('1110011001100110', 2), int('1000111100001100', 2)]

result = udp\_check\_sum(data\_list)

print('校验和为：' + np.binary\_repr(result))



**P2**

对于A而言，源IP为B，源端口号为80；目的IP为A，目的端口为26145。

对于C而言，源IP都为B，源端口号都为80；

左边的目的IP为C，目的端口号为7532。

右边的目的IP为C，目的端口号为26145。

**P4**

a. 00111110

b. 01000000

c. a中一、二字节变为01011101 、01100100