# ISCC_How_decode

```
  v2[10] = 0x5A630997;
  v2[11] = 0x80576CDB;
  v2[12] = 0x11783A4D;
  v2[13] = 0x73D2C70E;
  v2[14] = 0xD6EE81AC;
  v2[15] = 0xDAFA0F09;
  v2[16] = 0xAC79A9EC;
  v2[17] = 0x91F4B9B7;
  key[0] = 'I';
  key[1] = 'S';
  key[2] = 'C';
  key[3] = 'C';
  scanf("%s", flag);                            //
                                                // n = 18
                                                //
  n = strlen(flag);
  if ( n == n1 )
  {
    for ( i = 0; i < n; ++i )
      v2[i] = flag[i];
    encode(v2, n, key);
    for ( i_0 = 0; i_0 < n; ++i_0 )
    {
      if ( *(p_v1 + i_0) != v2[i_0] )
        goto LABEL_2;
    }
    printf("Your input is the right answer!");
    system("pause");
    return 0;
  }
  else
  {
LABEL_2:
    printf("Wrong answer!");
    system("pause");
    return 0;
  }
}
```

这个 encode 纠结了我好久

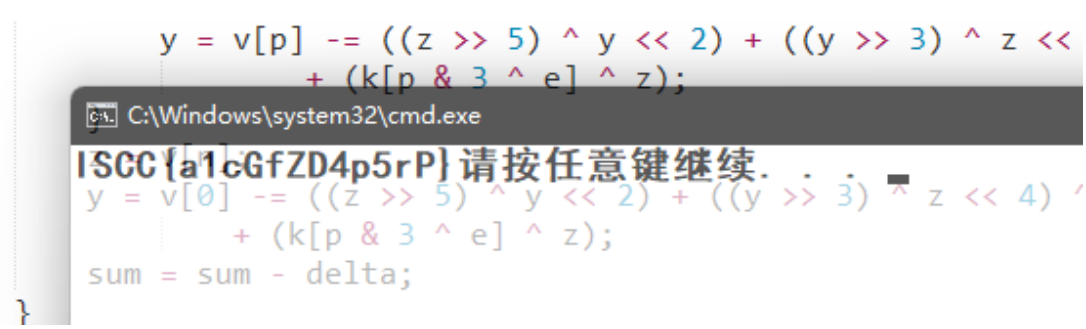
```
int p; // [rsp+14h] [rbp-Ch]
int sum; // [rsp+18h] [rbp-8h]
int z; // [rsp+1Ch] [rbp-4h]

rounds = 52 / n + 6;                            //
                                                // n = 18
                                                //
                                                // rounds = 8
sum = 0;
for ( z = v2[n - 1]; rounds--; z = *v5 )
{
  sum -= 0x61C88647;
  e = (sum >> 2) & 3;
  for ( p = 0; p < n - 1; ++p )
  {
    y = v2[p + 1];
    v4 = &v2[p];
    *v4 += ((y ^ sum) + (z ^ key[e ^ p & 3])) ^ (((4 * y) ^ (z >> 5)) + ((y >> 3) ^ (16 * z)));
    z = *v4;
  }
  v5 = &v2[n - 1];
  *v5 += ((*v2 ^ sum) + (z ^ key[e ^ p & 3])) ^ (((4 * *v2) ^ (z >> 5)) + ((*v2 >> 3) ^ (16 * z)));
}
}
```

学习之后知道了是 xxtea 加密害嗨嗨

学习 xxtea 之后了解了解密算法 开逆 开逆!!!!!

```
        v[1] = 0x15F437DE;
        v[2] = 0xEB4BF8AF;
        v[3] = 0xD9F98EF2;
        v[4] = 0x42CCAB39;
        v[5] = 0x7A857094;
        v[6] = 0x912E821D;
        v[7] = 0xCD3148B7;
        v[8] = 0x743BC712;
        v[9] = 0x487532A5;
        v[10] = 0x5A630997;
        v[11] = 0x80576CDB;
        v[12] = 0x11783A4D;
        v[13] = 0x73D2C70E;
        v[14] = 0xD6EE81AC;
        v[15] = 0xDAFA0F09;
        v[16] = 0xAC79A9EC;
        v[17] = 0x91F4B9B7;
        int n = 17;
        strcpy(k, "ISCC");
        int z = v[n], y = v[0], delta = 0x9E3779B9, sum, e;
        int p, q = 6 + 52 / (n + 1);
        sum = q * delta;
        while (sum != 0) {
            e = (sum >> 2) & 3;
            for (p = n; p > 0; p--) {
                z = v[p - 1];
                y = v[p] -= ((z >> 5) ^ y << 2) + ((y >> 3) ^ z << 4) ^ (sum ^ y)
                        + (k[p & 3 ^ e] ^ z);
            }
            z = v[n];
            y = v[0] -= ((z >> 5) ^ y << 2) + ((y >> 3) ^ z << 4) ^ (sum ^ y)
                    + (k[p & 3 ^ e] ^ z);
            sum = sum - delta;
        }
        for (int i = 0; i <= n; ++i )
                printf("%c", v[i]);
        return 0;
```




ISCC{a1cGfZD4p5rP}

这题目的考点主要是那个加密算法　没一下子解出来主要是经验不够 以后还得多多了解加密的算法