1、 下载一个 ida 版本，什么版本都可以，虽然本题是说在 Linux 下运行的文件，但实际上什么平台的 ida 都可以，还是 Windows 的好用，Linux 下我并没有能转成伪 C 代码。
2、 打开 rev1 文件。
3、 按下 F5 找到伪 C 代码。

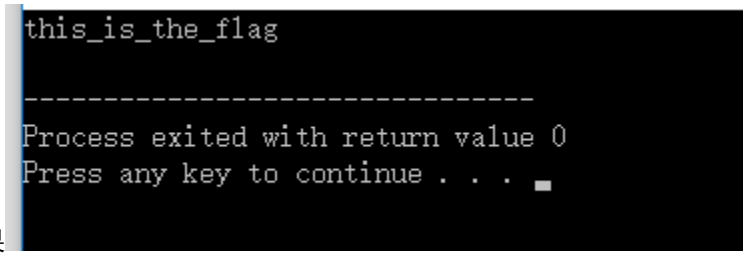
```
int __cdecl main(int argc, const char **argv, const char **envp)
{
  size_t v3; // rax@1
  int result; // eax@4
  __int64 v5; // rsi@4
  unsigned __int8 i; // [sp+7h] [bp-39h]@1
  _BYTE *ptr; // [sp+8h] [bp-38h]@1
  char s[8]; // [sp+10h] [bp-30h]@1
  __int64 v9; // [sp+28h] [bp-18h]@1

  v9 = *MK_FP(__FS__, 40LL);
  strcpy(s, "tikp[luX|aoTjaoh");
  v3 = strlen(s);
  ptr = malloc(v3);
  puts("Welcome!");
  puts("This is a x64 REV,find out the flag.");
  for ( i = 0; i < strlen(s); ++i )
  {
    ptr[i] = s[i] ^ i;
    ptr[i] = 0;
  }
  printf("CTF{%s}\n", ptr);
  free(ptr);
  result = 0;
  v5 = *MK_FP(__FS__, 40LL) ^ v9;
  return result;
}
```

可以看出关键部分在于字符串 s 和 for 循环的逻辑，可以看出 for 循环里的清零操作应该是误识别。

写出对应版本的转换代码

```
#include<iostream>
#include<string>
using namespace std;
int main()
{
    string str = "tikp[luX|aoTjaoh";
    for(int i = 0;i<str.length();++i)
    {
        str[i] = str[i]^i;
    }
    cout<<str<<endl;
}
```

得到结果


this_is_the_flag

Process exited with return value 0
Press any key to continue . . .