1.使用組合語言呼叫system call，從stdin讀進一個字元

新增一個ch 的字串，buffer size 為2

根據參數，編號0為 system call read ，將輸入結果丟到rsi

```c
#include <unistd.h>
#include <stdio.h>
#include <string.h>
int main(int argc, char ** argv)
{
    char ch[2]={};
    long len = 2;
    long ret;
    memset(ch,0,sizeof(ch));
    printf("please input a char\n");
    __asm__ volatile(
        "mov $0 ,%%rax\n"
        "mov $2 ,%%rdi\n"
        "mov %1 ,%%rsi\n"
        "mov %2 ,%%rdx\n"
        "syscall\n"
        "mov $1 ,%%rax\n"
        "mov $2 ,%%rdi\n"
        "mov %1 ,%%rsi\n"
        "mov %2 ,%%rdx\n"
        "syscall\n"
        "mov %%rax, %0"
        : "=m"(ret)
        : "g" (ch), "g" (len)
            : "rax", "rbx", "rcx", "rdx");
    printf("user input char %s\n",ch);
    printf("回傳值是：%ld\n", ret);

}
```

結果圖：

```
➜  ~ gcc syscallhw.c -o syscallhw
➜  ~ ./syscallhw
a
read char is a
```