Reverse Engineering Sprout 2018 Final Challenge (Nerfed)

```
rcs:

var_18= dword ptr -18h
var_14= dword ptr -14h
var_4= dword ptr -4

push    rbp
mov     rbp, rsp
mov     [rbp+var_14], edi
mov     [rbp+var_18], esi
mov     [rbp+var_4], 0
jmp     short loc_400663

loc_400659:
mov     eax, [rbp+var_14]
add     [rbp+var_4], eax
sub     [rbp+var_18], 1

loc_400663:
cmp     [rbp+var_18], 0
jnz     short loc_400659

mov     eax, [rbp+var_4]
pop     rbp
retn
rcs endp         ; function rcs end
```

```
main:

push    rbp
mov     rbp, rsp
sub     rsp, 20h
mov     rax, fs:28h
mov     [rbp+var_8], rax
xor     eax, eax
mov     edi, offset format ; "Enter UserID: "
mov     eax, 0
call    _printf
lea     rax, [rbp+var_14]
mov     rsi, rax
mov     edi, offset aD  ; "%d"
mov     eax, 0
call    ___isoc99_scanf
mov     edi, offset aEnterPasscode ; "Enter Passcode: "
mov     eax, 0
call    _printf
lea     rax, [rbp+var_10]
mov     rsi, rax
mov     edi, offset aD  ; "%d"
mov     eax, 0
call    ___isoc99_scanf
mov     eax, [rbp+var_14]
cmp     eax, 270Fh
jle     short loc_4006E1

mov     eax, [rbp+var_10]
cmp     eax, 63h
jg      short loc_4006F2

loc_4006E1:
mov     edi, offset s   ; "Numbers are too small."
call    _puts
mov     eax, 0
jmp     short loc_400748
```

```
loc_4006F2:
mov     eax, [rbp+var_10]
mov     edx, eax
mov     eax, [rbp+var_14]
mov     esi, edx
mov     edi, eax
call    rcs
mov     [rbp+var_C], eax
mov     eax, [rbp+var_10]
mov     edx, eax
mov     eax, [rbp+var_C]
mov     esi, edx
mov     edi, eax
call    rcs
mov     [rbp+var_C], eax
cmp     [rbp+var_C], 13B06165h
jnz     short loc_400739

mov     edi, offset aYouWin ; "You win!"
call    _puts
mov     edi, offset aFlagThePasswor ; "flag{xxx}"...
call    _puts
jmp     short loc_400743

loc_400739:
mov     edi, offset aWrongAnswer ; "Wrong Answer"
call    _puts

loc_400743:
mov     eax, 0

loc_400748:
mov     rcx, [rbp+var_8]
xor     rcx, fs:28h
jz      short locret_40075C

locret_40075C:
leave
retn
main endp
```