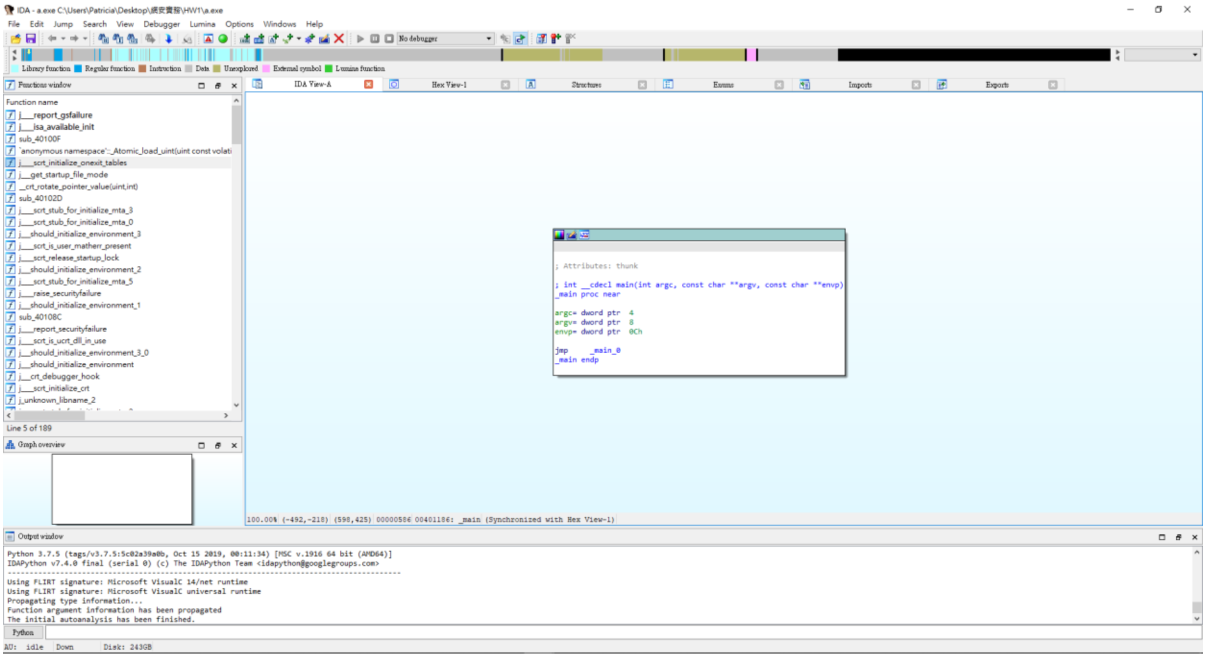
HW 1. Disassembly

L091025 清大資工所 熊詩旻

Q1. Disassemble and decompile a binary executable.

Steps:

1. Open the a.exe with IDA.



IDA - a.exe C:\Users\Patricia\Desktop\網安實務\HW1\a.exe
File Edit Jump Search View Debugger Lumina Options Windows Help
No debugger
Library function Regular function Instruction Data Unexplored External symbol Lumina function
Functions window
IDA View-A
Hex View-1
Structures
Enums
Imports
Exports
Line 5 of 189
Graph overview
Output window
Python
AU: idle Down Disk: 243GB

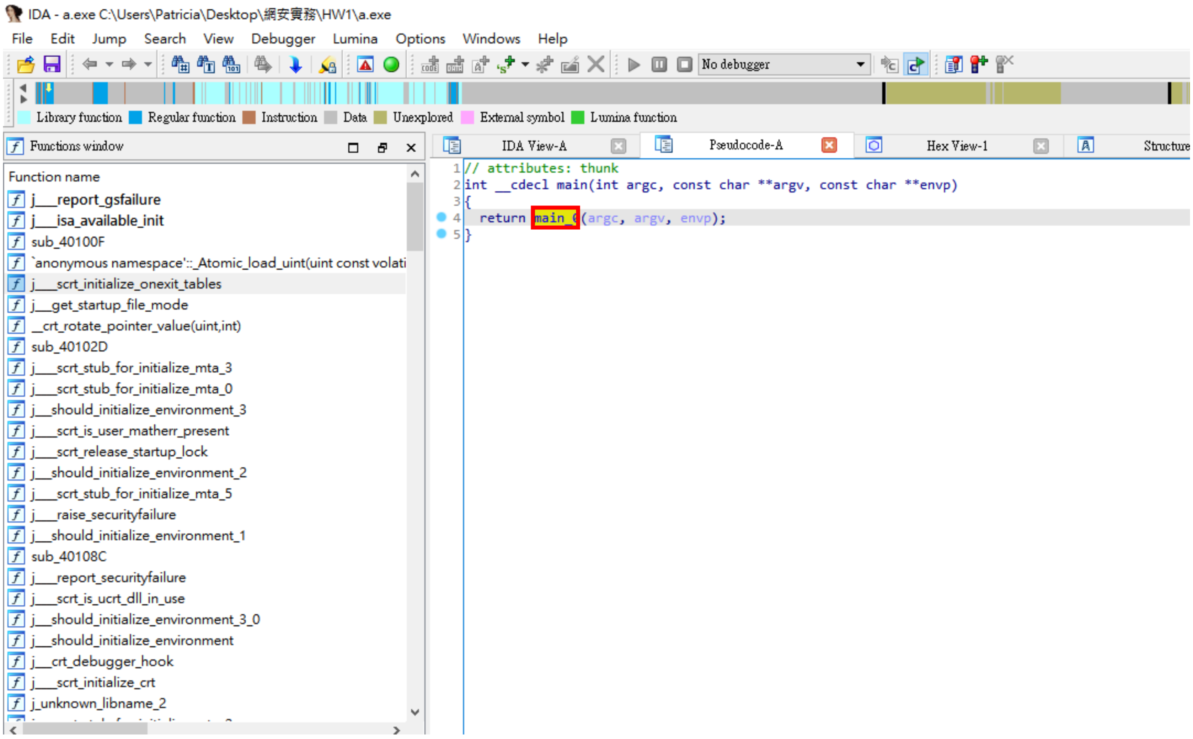
```
; Attributes: thunk

; int __cdecl main(int argc, const char **argv, const char **envp)
_main proc near

argc= dword ptr  4
argv= dword ptr  8
envp= dword ptr  0Ch

jmp     _main_0
_main endp
```

```
100.00% (-492,-218) (598,425) 00000586 00401186: _main (Synchronized with Hex View-1)
```

```
Python 3.7.5 (tags/v3.7.5:5c02a39a0b, Oct 15 2019, 00:11:34) [MSC v.1916 64 bit (AMD64)]
IDAPython v7.4.0 final (serial 0) (c) The IDAPython Team <idapython@googlegroups.com>
---------------------------------------------------------------------------------------
Using FLIRT signature: Microsoft VisualC 14/net runtime
Using FLIRT signature: Microsoft VisualC universal runtime
Propagating type information...
Function argument information has been propagated
The initial autoanalysis has been finished.
```

2. Press F5 to generate the pseudocode, double click the main.

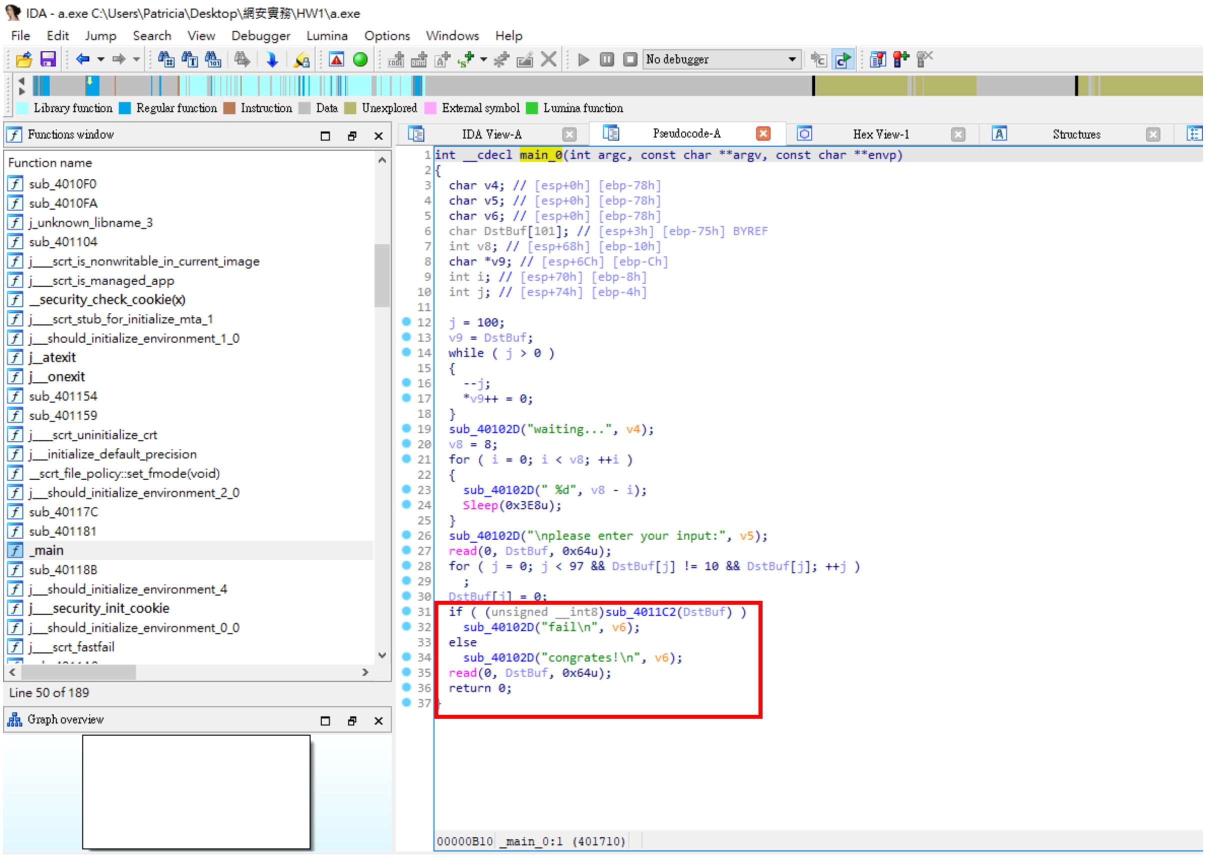
Then we find the main function.



IDA - a.exe C:\Users\Patricia\Desktop\網安實務\HW1\a.exe
File Edit Jump Search View Debugger Lumina Options Windows Help
No debugger
Library function Regular function Instruction Data Unexplored External symbol Lumina function
Functions window
IDA View-A
Pseudocode-A
Hex View-1
Structure

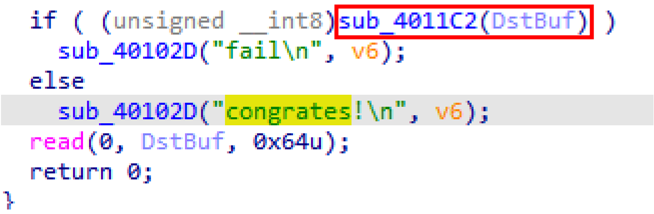
```
Function name
j___report_gsfailure
j___isa_available_init
sub_40100F
`anonymous namespace'::_Atomic_load_uint(uint const volati
j___scrt_initialize_onexit_tables
j___get_startup_file_mode
__crt_rotate_pointer_value(uint,int)
sub_40102D
j___scrt_stub_for_initialize_mta_3
j___scrt_stub_for_initialize_mta_0
j___should_initialize_environment_3
j___scrt_is_user_matherr_present
j___scrt_release_startup_lock
j___should_initialize_environment_2
j___scrt_stub_for_initialize_mta_5
j___raise_securityfailure
j___should_initialize_environment_1
sub_40108C
j___report_securityfailure
j___scrt_is_ucrt_dll_in_use
j___should_initialize_environment_3_0
j___should_initialize_environment
j___crt_debugger_hook
j___scrt_initialize_crt
j_unknown_libname_2
```

```
// attributes: thunk
int __cdecl main(int argc, const char **argv, const char **envp)
{
  return main_0(argc, argv, envp);
}
```


IDA - a.exe C:\Users\Patricia\Desktop\網安實務\HW1\a.exe
File Edit Jump Search View Debugger Lumina Options Windows Help
No debugger
Library function
Regular function
Instruction
Data
Unexplored
External symbol
Lumina function
Functions window
Function name
sub_4010F0
sub_4010FA
j_unknown_libname_3
sub_401104
j___scrt_is_nonwritable_in_current_image
j___scrt_is_managed_app
__security_check_cookie(x)
j___scrt_stub_for_initialize_mta_1
j___should_initialize_environment_1_0
j_atexit
j__onexit
sub_401154
sub_401159
j___scrt_uninitialize_crt
j__initialize_default_precision
__scrt_file_policy::set_fmode(void)
j___should_initialize_environment_2_0
sub_40117C
sub_401181
_main
sub_40118B
j___should_initialize_environment_4
j___security_init_cookie
j___should_initialize_environment_0_0
j___scrt_fastfail
Line 50 of 189
Graph overview
IDA View-A
Pseudocode-A
Hex View-1
Structures
00000B10 _main_0:1 (401710)

```
int __cdecl main_0(int argc, const char **argv, const char **envp)
{
  char v4; // [esp+0h] [ebp-78h]
  char v5; // [esp+0h] [ebp-78h]
  char v6; // [esp+0h] [ebp-78h]
  char DstBuf[101]; // [esp+3h] [ebp-75h] BYREF
  int v8; // [esp+68h] [ebp-10h]
  char *v9; // [esp+6Ch] [ebp-Ch]
  int i; // [esp+70h] [ebp-8h]
  int j; // [esp+74h] [ebp-4h]

  j = 100;
  v9 = DstBuf;
  while ( j > 0 )
  {
    --j;
    *v9++ = 0;
  }
  sub_40102D("waiting...", v4);
  v8 = 8;
  for ( i = 0; i < v8; ++i )
  {
    sub_40102D(" %d", v8 - i);
    Sleep(0x3E8u);
  }
  sub_40102D("\nplease enter your input:", v5);
  read(0, DstBuf, 0x64u);
  for ( j = 0; j < 97 && DstBuf[j] != 10 && DstBuf[j]; ++j )
    ;
  DstBuf[j] = 0;
  if ( (unsigned __int8)sub_4011C2(DstBuf) )
    sub_40102D("fail\n", v6);
  else
    sub_40102D("congrates!\n", v6);
  read(0, DstBuf, 0x64u);
  return 0;
}
```

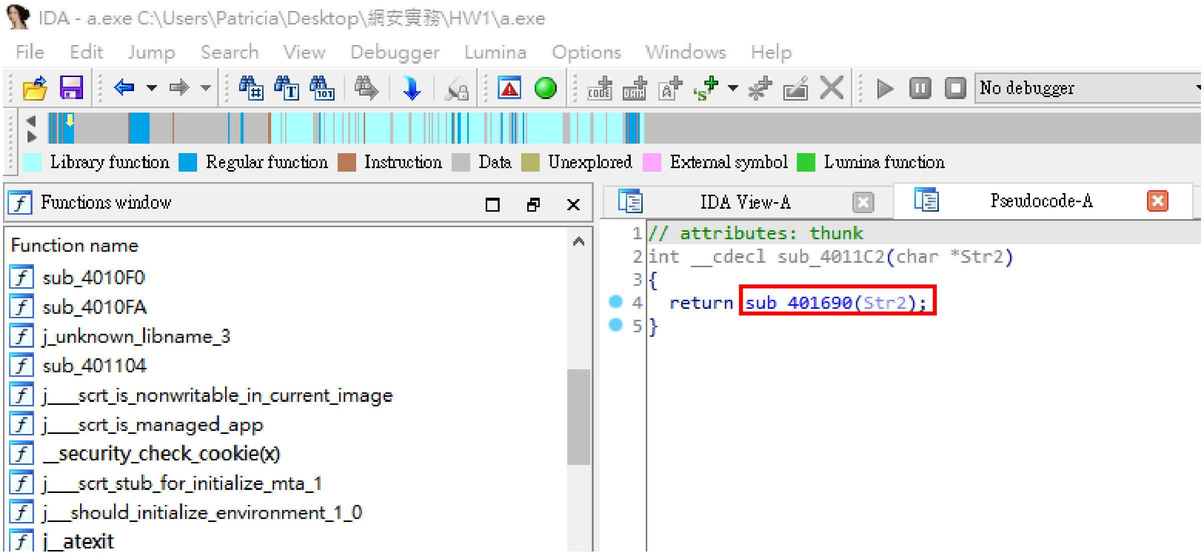
3. Observe the function, we can find there is a “fail\n” and “congrates!\n” message.



```
  if ( (unsigned __int8)sub_4011C2(DstBuf) )
    sub_40102D("fail\n", v6);
  else
    sub_40102D("congrates!\n", v6);
  read(0, DstBuf, 0x64u);
  return 0;
}
```

4. Observe the function, we know that if we want a congrats message, we need sub_4011C2() function to be FALSE.

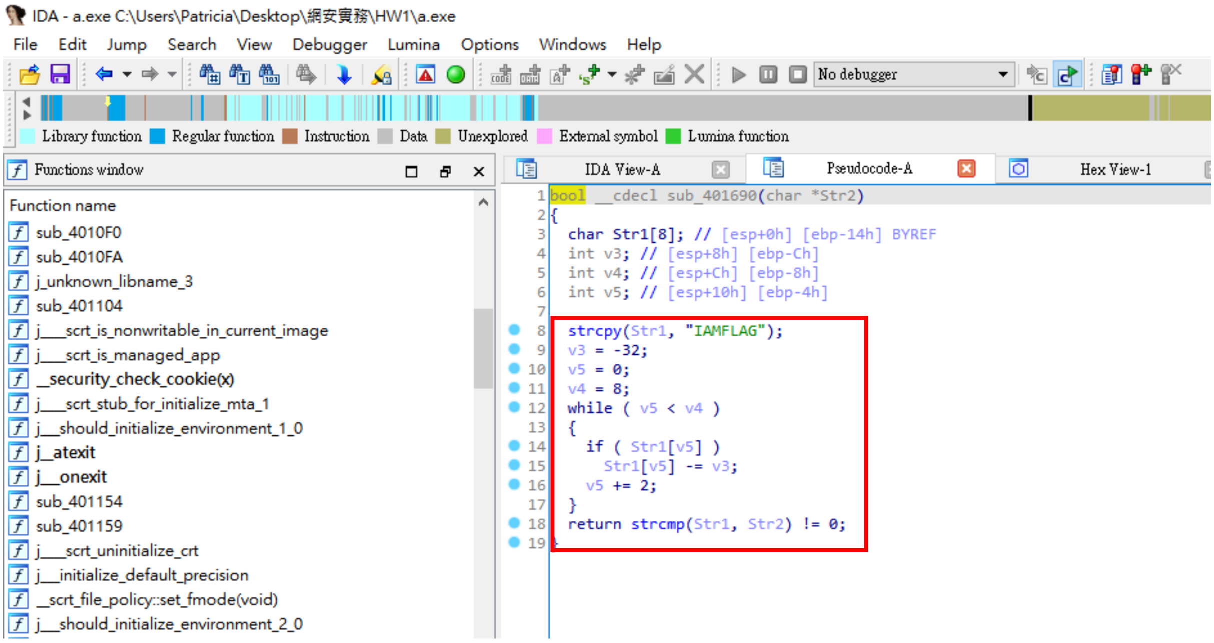
   We double click the sub_4011C2() function to see what it will do.


IDA - a.exe C:\Users\Patricia\Desktop\網安實務\HW1\a.exe
File Edit Jump Search View Debugger Lumina Options Windows Help
No debugger
Library function
Regular function
Instruction
Data
Unexplored
External symbol
Lumina function
Functions window
Function name
sub_4010F0
sub_4010FA
j_unknown_libname_3
sub_401104
j___scrt_is_nonwritable_in_current_image
j___scrt_is_managed_app
__security_check_cookie(x)
j___scrt_stub_for_initialize_mta_1
j___should_initialize_environment_1_0
j_atexit
IDA View-A
Pseudocode-A

```
// attributes: thunk
int __cdecl sub_4011C2(char *Str2)
{
  return sub_401690(Str2);
}
```

We find it returns a value from another function sub_401690(), double click to take a look.

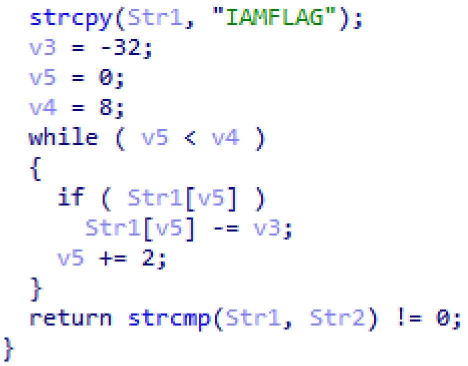
5. We can see that the function will return a Boolean value, and we find a string “IAMFLAG” seems to be our password, but there looks like something will done with it. Observe the follow code to see what happen.


IDA - a.exe C:\Users\Patricia\Desktop\網安實務\HW1\a.exe
File Edit Jump Search View Debugger Lumina Options Windows Help
No debugger
Library function Regular function Instruction Data Unexplored External symbol Lumina function
Functions window
IDA View-A
Pseudocode-A
Hex View-1

Function name
sub_4010F0
sub_4010FA
j_unknown_libname_3
sub_401104
j___scrt_is_nonwritable_in_current_image
j___scrt_is_managed_app
__security_check_cookie(x)
j___scrt_stub_for_initialize_mta_1
j__should_initialize_environment_1_0
j_atexit
j__onexit
sub_401154
sub_401159
j___scrt_uninitialize_crt
j__initialize_default_precision
__scrt_file_policy::set_fmode(void)
j__should_initialize_environment_2_0

```
bool __cdecl sub_401690(char *Str2)
{
  char Str1[8]; // [esp+0h] [ebp-14h] BYREF
  int v3; // [esp+8h] [ebp-Ch]
  int v4; // [esp+Ch] [ebp-8h]
  int v5; // [esp+10h] [ebp-4h]

  strcpy(Str1, "IAMFLAG");
  v3 = -32;
  v5 = 0;
  v4 = 8;
  while ( v5 < v4 )
  {
    if ( Str1[v5] )
      Str1[v5] -= v3;
    v5 += 2;
  }
  return strcmp(Str1, Str2) != 0;
}
```

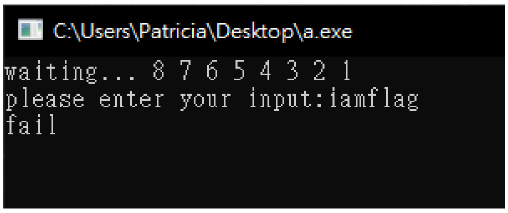
6. Recall that we need the function to return FALSE, so the last function strcmp() need to be FALSE, Means that Str1 need to be the same as Str2. Because Str2 is our input, so now we have to find the Str1.



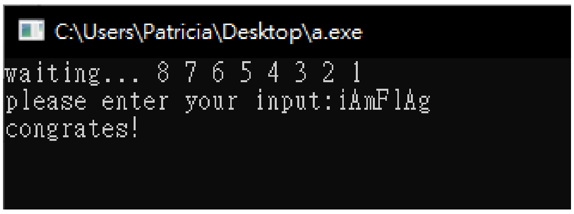
```
  strcpy(Str1, "IAMFLAG");
  v3 = -32;
  v5 = 0;
  v4 = 8;
  while ( v5 < v4 )
  {
    if ( Str1[v5] )
      Str1[v5] -= v3;
    v5 += 2;
  }
  return strcmp(Str1, Str2) != 0;
}
```

First we can see that Str1 is copy from “IAMFLAG”, then the while loop will add 32 to every two char, means that turn it into lower case. So we get our password “iAmFlAg”.

7. Test with the a.exe.



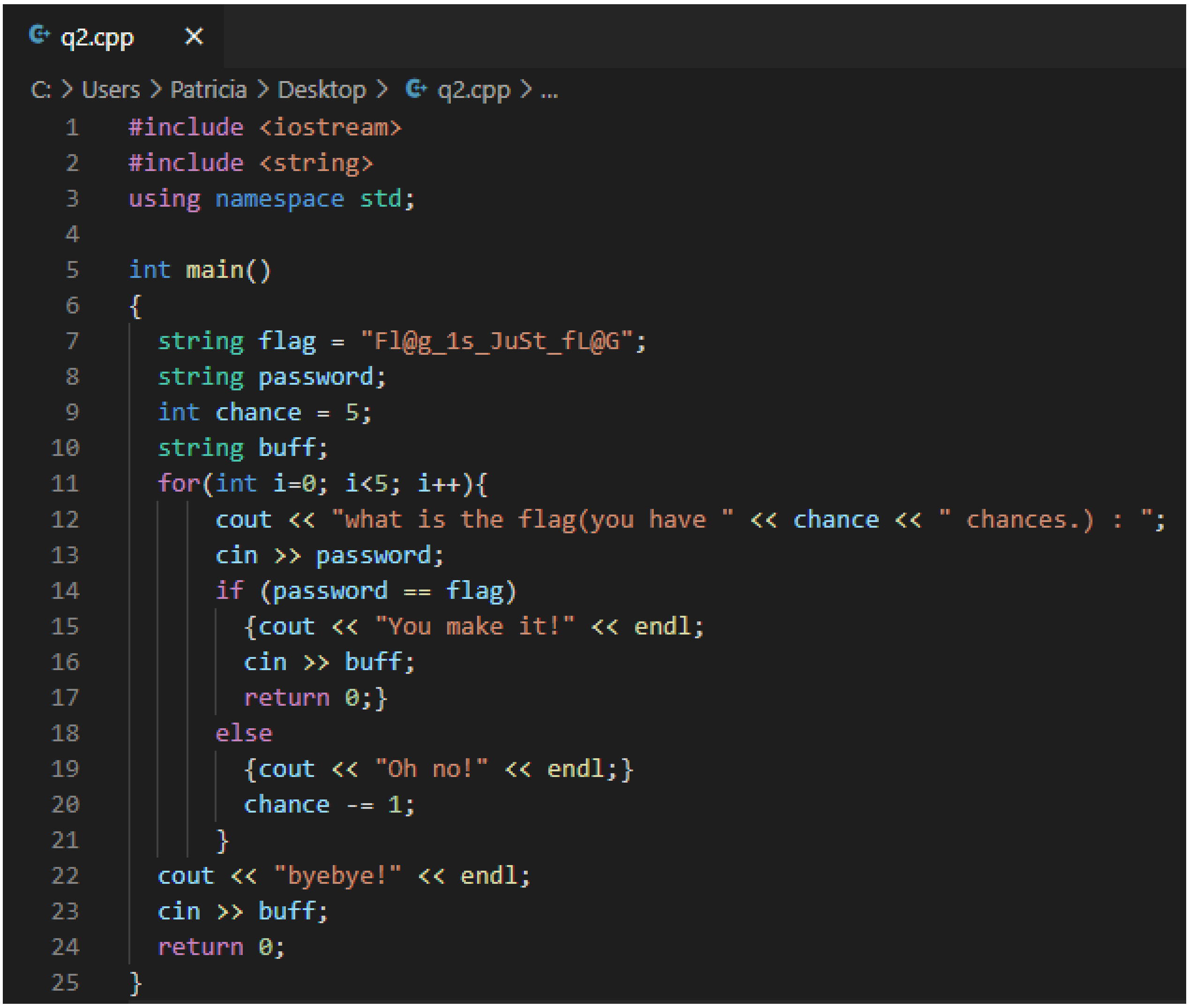
```
C:\Users\Patricia\Desktop\a.exe
waiting... 8 7 6 5 4 3 2 1
please enter your input:iamflag
fail
```



```
C:\Users\Patricia\Desktop\a.exe
waiting... 8 7 6 5 4 3 2 1
please enter your input:iAmFlAg
congrates!
```

Q2. Obfuscation observation.

1. My Source code.



```cpp
#include <iostream>
#include <string>
using namespace std;

int main()
{
  string flag = "Fl@g_1s_JuSt_fL@G";
  string password;
  int chance = 5;
  string buff;
  for(int i=0; i<5; i++){
      cout << "what is the flag(you have " << chance << " chances.) : ";
      cin >> password;
      if (password == flag)
        {cout << "You make it!" << endl;
        cin >> buff;
        return 0;}
      else
        {cout << "Oh no!" << endl;}
        chance -= 1;
      }
  cout << "byebye!" << endl;
  cin >> buff;
  return 0;
}
```