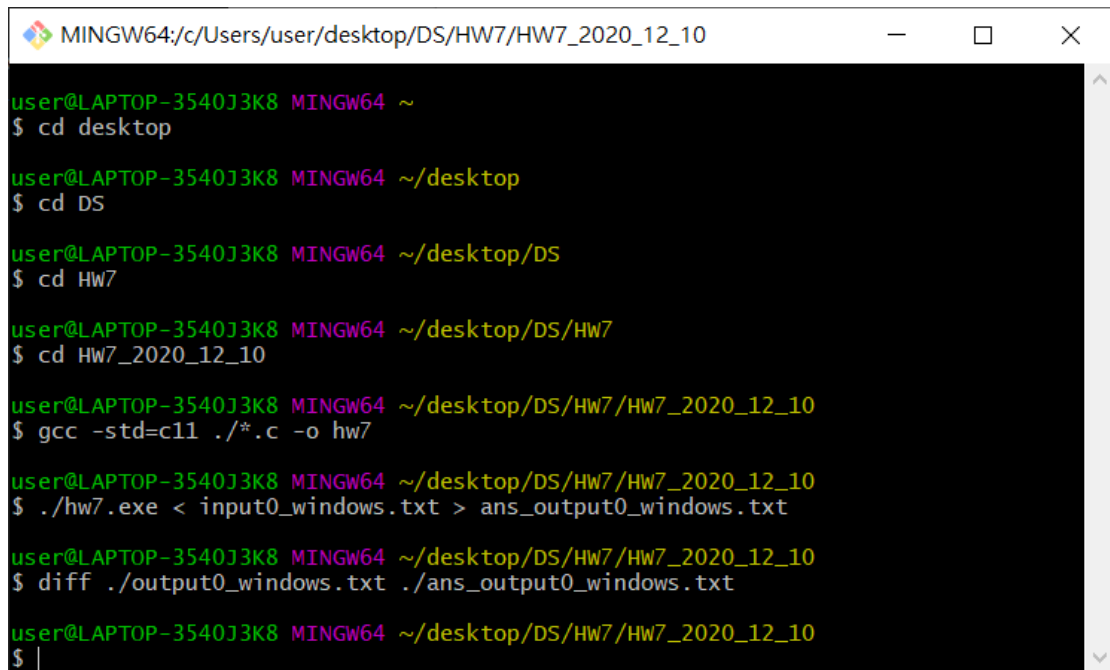


Result Screenshots



```
MINGW64:/c:/Users/user/desktop/DS/HW7/HW7_2020_12_10
user@LAPTOP-3540J3K8 MINGW64 ~
$ cd desktop

user@LAPTOP-3540J3K8 MINGW64 ~/desktop
$ cd DS

user@LAPTOP-3540J3K8 MINGW64 ~/desktop/DS
$ cd HW7

user@LAPTOP-3540J3K8 MINGW64 ~/desktop/DS/HW7
$ cd HW7_2020_12_10

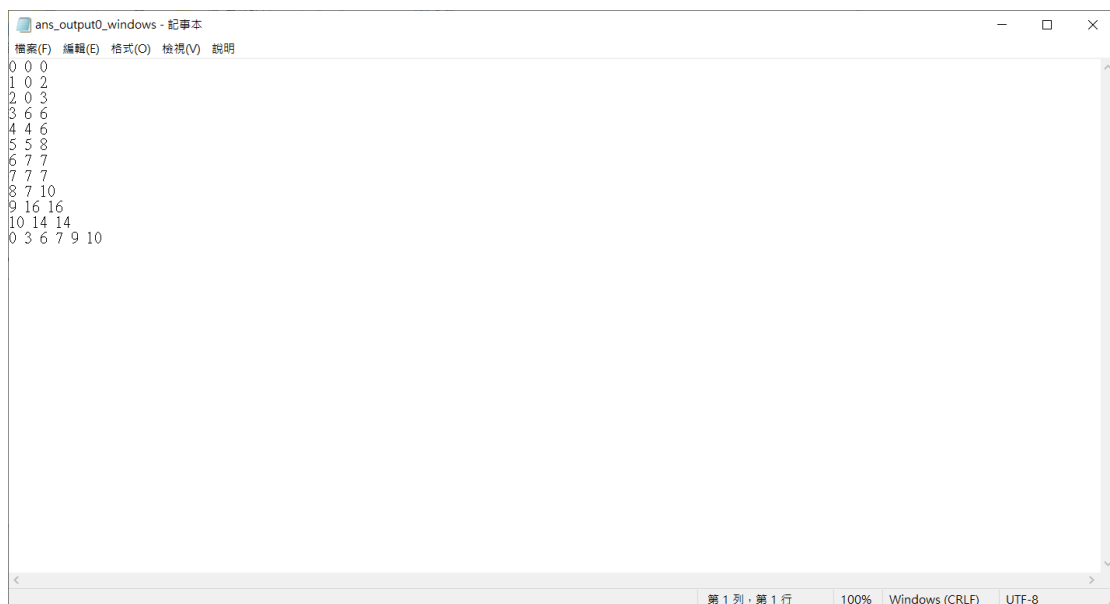
user@LAPTOP-3540J3K8 MINGW64 ~/desktop/DS/HW7/HW7_2020_12_10
$ gcc -std=c11 ./*.c -o hw7

user@LAPTOP-3540J3K8 MINGW64 ~/desktop/DS/HW7/HW7_2020_12_10
$ ./hw7.exe < input0_windows.txt > ans_output0_windows.txt

user@LAPTOP-3540J3K8 MINGW64 ~/desktop/DS/HW7/HW7_2020_12_10
$ diff ./output0_windows.txt ./ans_output0_windows.txt

user@LAPTOP-3540J3K8 MINGW64 ~/desktop/DS/HW7/HW7_2020_12_10
$ |
```

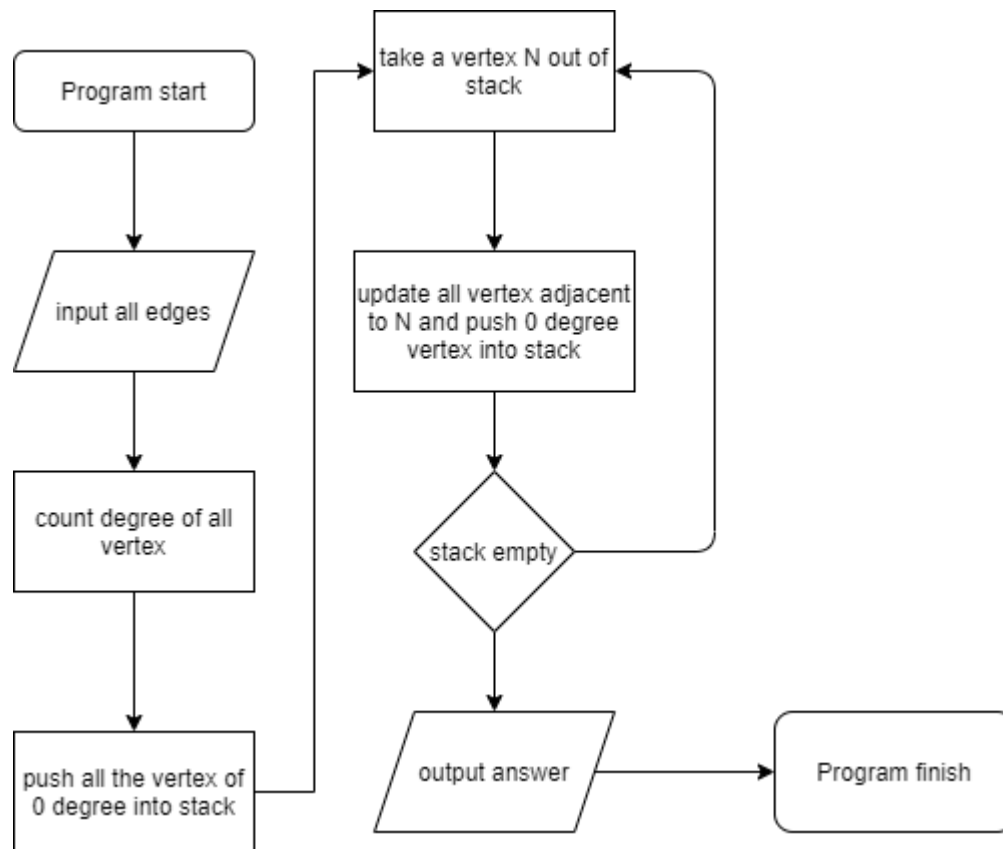
編譯與執行指令截圖



```
ans_output0_windows - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明
0 0 0
1 0 2
2 0 3
3 6 6
4 4 6
5 5 8
6 7 7
7 7 7
8 7 10
9 16 16
10 14 14
0 3 6 7 9 10
```

ans_output0_windows.txt 截圖

Program Architecture



Program Functions

The program has no function except main function

Program design

先輸入所有邊的資訊

存成 **adjacent list** 以及 **adjacent matrix**

統計每個節點的入度，將入度為 0 的節點壓進 **stack**

每次從 **stack** 取出一個節點 **n**，直到 **stack** 為空為止

然後以 **n** 進行 **relax**，對相鄰的邊做修正

For all **i** (**n** 下一個可以走到 **i**) $\text{early}(\mathbf{i}) = \max(\text{early}(\mathbf{i}), \text{early}(\mathbf{n}) + E(\mathbf{n}, \mathbf{i}))$

要求 **late** 則是反過來以同樣的方法從中點開始求

Operating System

Windows10 家用版

Compiler

gcc version 8.2.0 (MinGW.org GCC-8.2.0-3)

Compile

```
gcc -std=c11 ./*.c -o hw7
```

Run

```
./hw7.exe < input0_windows.txt > ans_output0_windows.txt
```