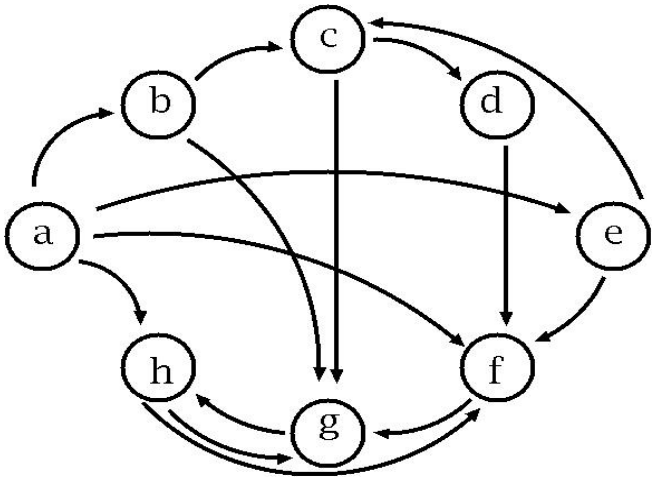
Exercise 6. Answer Sheet

Student's Name: ____Tran Thi Thoa_____ Student's ID: _____s1242006_____

Problem 1. Given the graph below



a) *(10 points)* Fill the following matrix by putting 1 if there is an edge between nodes. Put 0 otherwise.

	a	b	c	d	e	f	g	h
a	1	1	0	0	1	1	0	1
b	0	1	1	0	0	0	1	0
c	0	0	1	1	0	0	1	0
d	0	0	0	1	0	1	0	0
e	0	0	1	0	1	1	0	0
f	0	0	0	0	0	1	1	0
g	0	0	0	0	0	0	1	1
h	0	0	0	0	0	1	1	1

b) *(40 points)* Write a program implementing Warshal's algorithm. Upload your code. Use your program to create a transitive closure G* of the graph above and show it in the space below.

Transitive closure defined by adjacency table

	a	b	c	d	e	f	g	h
a	1	1	1	1	1	1	1	1
b	0	1	1	1	0	1	1	1
c	0	0	1	1	0	1	1	1
d	0	0	0	1	0	1	1	1
e	0	0	1	1	1	1	1	1
f	0	0	0	0	0	1	1	1
g	0	0	0	0	0	1	1	1
h	0	0	0	0	0	1	1	1

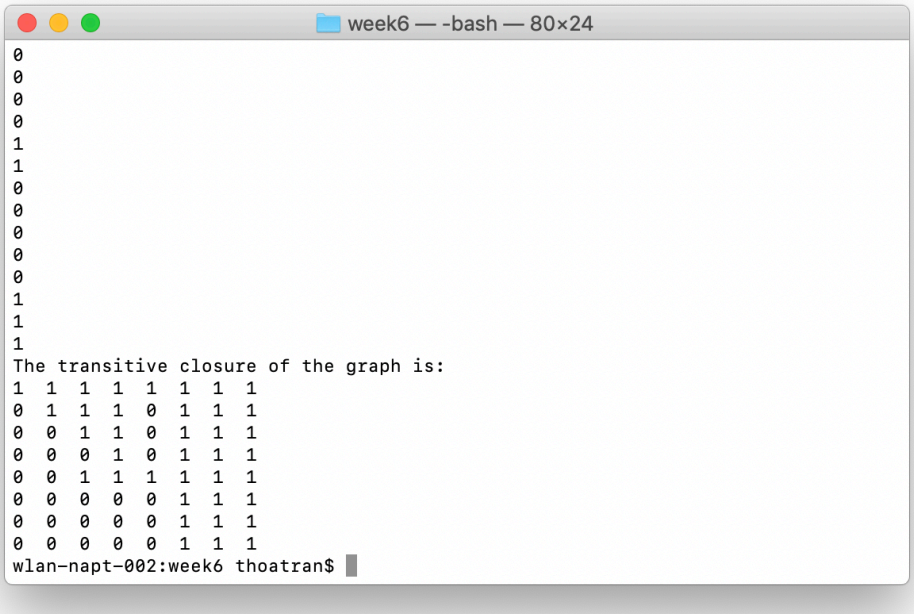
The implementation of Warshall's algorithm in C++

Input: firstly input the number of rows(equal to the number of columns) in the adjacent matrix then input each cell in that matrix following the rule of the warshall's algorithm

Output: the transitive closure matrix of the graph

To compile and run file, run the following command lines

```
g++ -std=c++11 -o warshall.o warshall.cpp
./warshall.o
```



Problem 2. *(50 points)* Consider the following weight adjacency matrix.

	a	b	c	d	e	f	g	h
a	0	48	∞	8	20	∞	20	∞
b	∞	0	24	∞	9	∞	76	29
c	97	∞	0	∞	∞	∞	18	1
d	∞	52	34	0	29	∞	∞	∞
e	∞	∞	∞	∞	0	10	∞	∞
f	∞	10	85	43	∞	0	41	29
g	∞	∞	∞	76	38	∞	0	∞
h	28	42	∞	77	21	∞	11	0

Write a program implementing Floyd's algorithm. Upload your code. Given the matrix above, calculate all pairs shortest paths using your program and fill the table below:

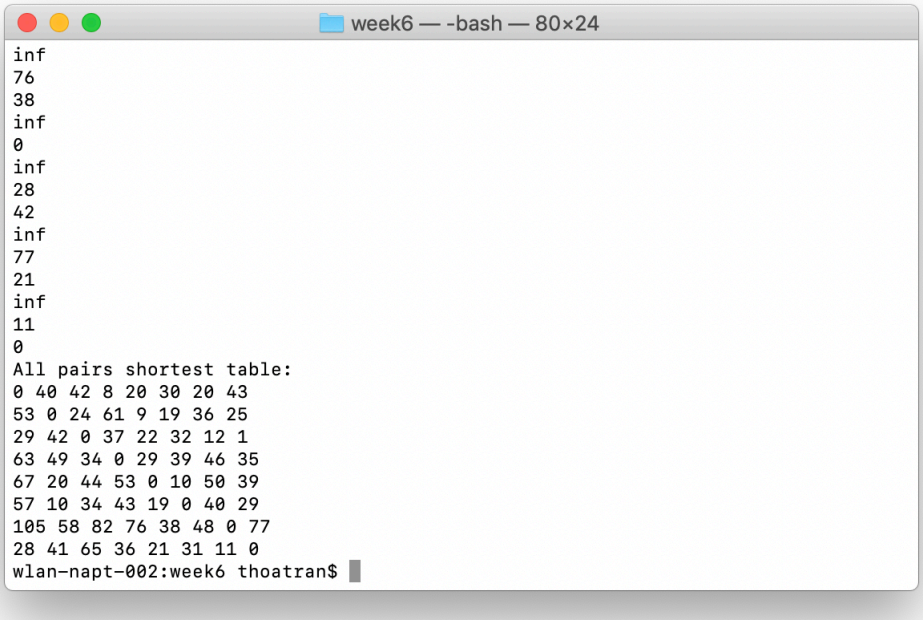
All pairs shortest path table

	a	b	c	d	e	f	g	h
a	0	40	42	8	20	30	20	43
b	53	0	24	61	9	19	36	25
c	29	42	0	37	22	32	12	1
d	63	49	34	0	29	39	46	35
e	67	20	44	53	0	10	50	39
f	57	10	34	43	19	0	40	29
g	105	58	82	76	38	48	0	77
h	28	41	65	36	21	31	11	0

The Floyd's algorithm is implemented in C++ in the file floyd.cpp
To compile and run the file, change the directory to the folder where you saved it and run the following command line:

```
g++ -std=c++11 -o floyd.o floyd.cpp
./floy.o
```

Input: first input the number of rows (equal to the number of columns) of the weight matrix.
Then input the value of each cell in the weight matrix respectively.
Note: For the vertex(cell) has weight equal to infinite, input "inf" to represent it.
The output will be the APSP matrix D (the weight matrix of all pairs with the shortest path)


week6 — -bash — 80×24
inf
76
38
inf
0
inf
28
42
inf
77
21
inf
11
0
All pairs shortest table:
0 40 42 8 20 30 20 43
53 0 24 61 9 19 36 25
29 42 0 37 22 32 12 1
63 49 34 0 29 39 46 35
67 20 44 53 0 10 50 39
57 10 34 43 19 0 40 29
105 58 82 76 38 48 0 77
28 41 65 36 21 31 11 0
wlan-napt-002:week6 thoatran$