KMIT – ARJUNA

Season-4

KMIT-APA-4004 **Programming Assignments** **Sunday 15th Sept, 2019**

1 Chocolate Eating Competition

Madhu (Maddy) madly loves chocolates. She came across an advertisement about *Chocolate Eating* competition. In the competition, some random number of chocolates (of similar type) would be placed in some random number of bowls at random positions in a row. The rules of the competition are as follows:
Each time, the participant must start eating chocolates by picking the current top two bowls i.e., more number of chocolates (say A and B). If both A and B have the same number of chocolates then eat chocolates and remove both A and B. If not equal then the participant must eat only smaller number of chocolates from both bowls and remove only that bowl. This process continues until only one bowl is left with few or no chocolates in it.

The prize is $100 times the number of chocolates that remain towards the end.

With the given number of bowls and chocolates in them, Maddy wants to know how many chocolates would remain in the last bowl so as to motivate herself to participate in the competition. Please help her with a program.

Input/Output

Input	Output	Comments
9 9 10 25 12	3	● First line 9 9 10 25 12 corresponds to the number of chocolates in those 5 bowls. ● Maddy must ○ First pick bowls with 25 and 12 chocolates (say A and B). Eat **12** chocolates each from both A and B. Remove B. A now contains 13 chocolate. **[9 9 10 13]** ○ Next pick bowls with 13 and 10 chocolates (say A and B). Eat **3** chocolates each from both A and B. Remove B. A now contains 3 chocolates. **[9 9 3]** ○ Next pick bowls with 9 and 9 chocolates (say A and B). Eat 9 chocolates from both A and B. Remove A and B because they are empty. [**3]** ○ Only bowl remains with one chocolate. Hence, the output is **3.**
5	**5**	if there is only one bowl then the output is chocolates contained in it, so **5**

2 Tom’s Trap for Jerry

Tom designed a trap of N x N dimensions to catch Jerry. At random positions (or cell) of the trap, there are two types of things. (1) A rat catching glue sheet (**represented as 0**). (2) A random number of cheese pieces. *An empty cel*l i.e., a cell neither has glue sheet nor cheese pieces is **represented as 1.**

Tom challenged Jerry to start from first cell (0, 0) and eat all the pieces of cheese starting from the cell with the least number of cheese pieces without stepping on any of the glue sheets. Tom told Jerry that **no two cells shall have equal number of cheese pieces**, and he would be generous enough to always place some cheese pieces to eat and **never fill the trap with glue sheets only.**

Now, write a program for Jerry to find out the minimum number of moves he must make to finish eating all pieces of cheese from all cells (with cheese) of the trapping grid. Return **-1** if it is not possible to eat all pieces of cheese from all cells.

Input/Output

Input	Output	Comments
3 **1 4 6** **0 0 7** **0 9 8**	**5**	● First line 3 corresponds to the size of the trap grid ● Next three lines represent the composition of trap grid 1 4 6 – (1) empty cell, 4 and 6 cheese pieces in Cell 2 & 3 0 0 7 – (0 0) two rat trapping glue sheets and four cheese pieces 0 9 8 – (0) one rat trapping glue 9 and 8 cheese pieces ● Cell 2 has the least number of cheese pieces. So, start eating from cell 2. ● Jerry can take steps in **1->4->6->7->8->9** order. It can eat all pieces in 5 moves.
3 **1 8 0** **2 0 5** **6 3 0**	**-1**	● Jerry can’t eat cheese pieces from cell 5. So, it **cannot** eat all cheese pieces from all cells. (-1)
3 **3 2 8** **0 0 4** **7 6 5**	**12**	● Jerry starts eating cheese pieces from cell 2 because it has the least number of cheese pieces **(3->2**). after eating from cell 2 it moves towards cell 1 and eat 3 cheese pieces **(2->3).** Then it moves towards 4 **(3->2->8->4)** followed by **(4->5), (5->6), (6->7), (7->6->5->4->8)** ● So in all Jerry must take **12** moves to finish eating all the cheese pieces from all plates without stepping on any of the glue sheet.

3 Distribution of Electronic Goods

Arjunasia, an electronic goods manufacturing company has identified some cities, connected by roads, for distribution of their products. The cities are assigned of IDs starting from 1. Based on the road connectivity among the cities and the distance between them, the company wants to identify a city that can become their distribution hub to supply goods to all other cities.

The company assumes:
1) All transport vehicles run at similar speed and take an equal amount of time to travel the same distance.

2) From a selected city, all transport vehicles start their journey to other cities at the same time.

3) Consider only one way of the roads connecting cities, as products must be delivered to other cities

Given the number of cities, their interconnectivity and the distance between them, write a program to find out the maximum time taken to reach all cities from a given city. Return -1 if goods cannot be sent to all cities.

Input/Output

Input	Output	Comments
4 3 **3 4 1** **2 1 2** **2 3 3** **2**	**4**	● First line **4** corresponds to the total number of cities, **3** corresponds to the total number of connections ● Each of the next three represents, City ID, City ID, Distance **3 4 1 - City 3 and City 4 are connected, distance between them is 1** **2 1 2 - City 2 and City 1 are connected, distance between them is 2** **2 3 3 - City 2 and City 3 are connected, distance between them is 3** ● Last line **2** corresponds to selected City ID to be a distribution hub ● Time to reach city 1 from 2 is **2** ● Time to reach city 3 from 2 is **3** ● Time to reach city 4 from 3 is **1** ● **Total 2 routes from city-2 to reach all other cities** ● **Route 1: 2->3->4 (3+1=4) and Route 2: 2->1 is 2** ● **Max of two routes is 4**
4 3 **1 2 5** **2 3 6** **1 4 10** **4**	**-1**	There is no route from city **4** to reach all cities, so output **-1**.