

ACM-ICPC and you

Daniel Epstein, CSE Grad Student

10/10/16



About me

- PhD Student in CSE
 - research in Human-Computer Interaction
- Undergrad degree from University of Virginia
- Competed in ICPC as an undergrad, went to ICPC World Finals in 2011

This year's contest

- Saturday, October 22nd, 9:00am-4:00pm
- Start in EEB-105, contest in CSE 003
- (tell your friends)

Why you should come

- Free food
- Hang out and code with friends
- Great prizes from Google
- Practice problem-solving skills
- You get to write bad code and nobody will yell at you

What are programming
competitions?





TEAM # 14

TEAM # 10

A group of students are working on computers in a classroom setting. In the foreground, a young man on the left gives a thumbs-up, while a young man in the center and a young woman on the right smile at the camera. They are all wearing white t-shirts with a logo and name tag. Other students are visible in the background, focused on their work. The room has white walls and a long desk.

3 people!

A group of students are working on computers in a classroom setting. In the foreground, a young man is smiling and giving a thumbs-up. Next to him, a young woman is also smiling. They are both looking towards the camera. In the background, several other students are seated at their desks, focused on their work. The room has white walls and a long desk. The text "1 computer!" is overlaid in large, blue, sans-serif font.

1 computer!



4065 - Rock, Paper, or Scissors?

North America - Pacific Northwest - 2007/2008

Rock, Paper, Scissors is a two player game, where each player simultaneously chooses one of the three items after counting to three. The game typically lasts a pre-determined number of rounds. The player who wins the most rounds wins the game. Given the number of rounds the players will compete, it is your job to determine which player wins after those rounds have been played.

The rules for what item wins are as follows:

- Rock always beats Scissors (Rock crushes Scissors)
- Scissors always beat Paper (Scissors cut Paper)
- Paper always beats Rock (Paper covers Rock)

Input

The first value in the input file will be an integer t ($0 < t < 1000$) representing the number of test cases in the input file. Following this, on a case by case basis, will be an integer n ($0 < n < 100$) specifying the number of rounds of Rock, Paper, Scissors played. Next will be n lines, each with either a capital R, P, or S, followed by a space, followed by a capital R, P, or S, followed by a newline. The first letter is Player 1's choice; the second letter is Player 2's choice.

Output

For each test case, report the name of the player ('Player 1' or 'Player 2') that wins the game, followed by a newline. If the game ends up in a tie, print 'TIE'.

Sample Input

```
3
2
R P
S R
3
P P
R S
S R
1
P R
```

Sample Output

```
Player 2
TIE
Player 1
```



4065 - Rock, Paper, or Scissors?

North America - Pacific Northwest - 2007/2008

Rock, Paper, Scissors is a two player game, where each player simultaneously chooses one of the three items after counting to three. The game typically lasts a pre-determined number of rounds. The player who wins the most rounds wins the game. Given the number of rounds the players will compete, it is your job to determine which player wins after those rounds have been played.

The rules for what item wins are as follows:

- Rock always beats Scissors (Rock crushes Scissors)
- Scissors always beat Paper (Scissors cut Paper)
- Paper always beats Rock (Paper covers Rock)

Input

The first value in the input file will be an integer t ($0 < t < 1000$) representing the number of test cases in the input file. Following this, on a case by case basis, will be an integer n ($0 < n < 100$) specifying the number of rounds of Rock, Paper, Scissors played. Next will be n lines, each with either a capital R, P, or S, followed by a space, followed by a capital R, P, or S, followed by a newline. The first letter is Player 1's choice; the second letter is Player 2's choice.

Output

For each test case, report the name of the player ('Player 1' or 'Player 2') that wins the game, followed by a newline. If the game ends up in a tie, print 'TIE'.

Sample Input

```
3
2
R P
S R
3
P P
R S
S R
1
P R
```

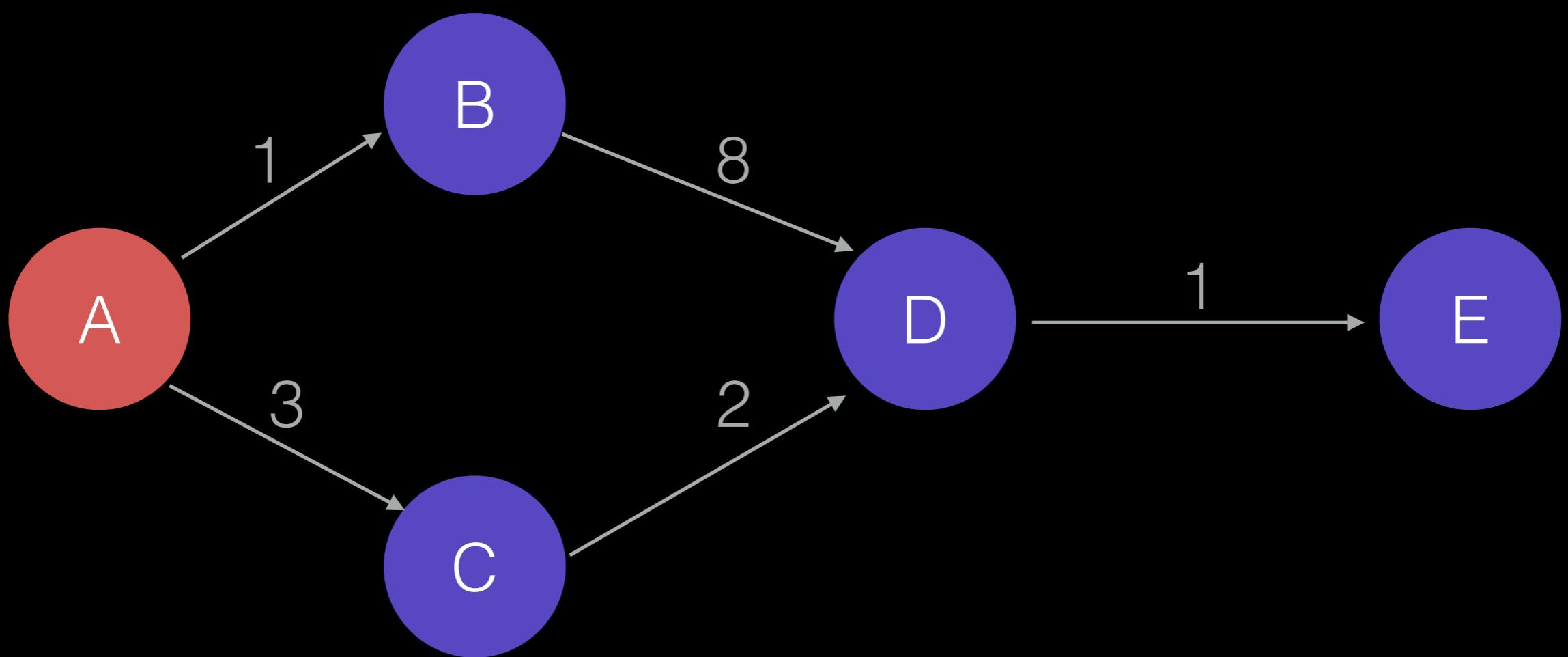
Sample Output

```
Player 2
TIE
Player 1
```

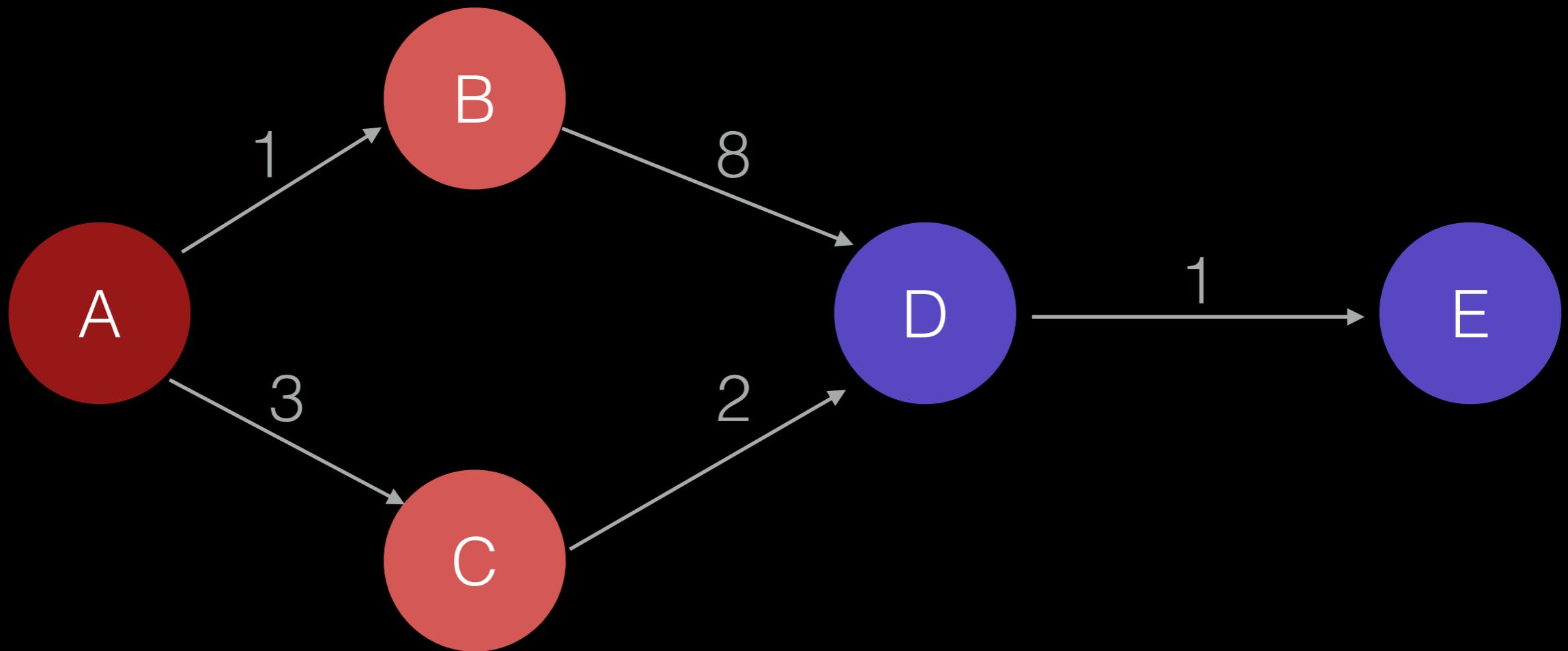
```
import java.util.*;
import java.io.*;

public class Main {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        int cases = in.nextInt();
        for(int i=0;i<cases;i++) {
            int n = in.nextInt();
            int d = in.nextInt();
            int count = 0;
            for(int j=0;j<n;j++) {
                int v = in.nextInt();
                int f = in.nextInt();
                int c = in.nextInt();
                if((v*f)/c >= d) {
                    count++;
                }
            }
            System.out.println(count);
        }
    }
}
```

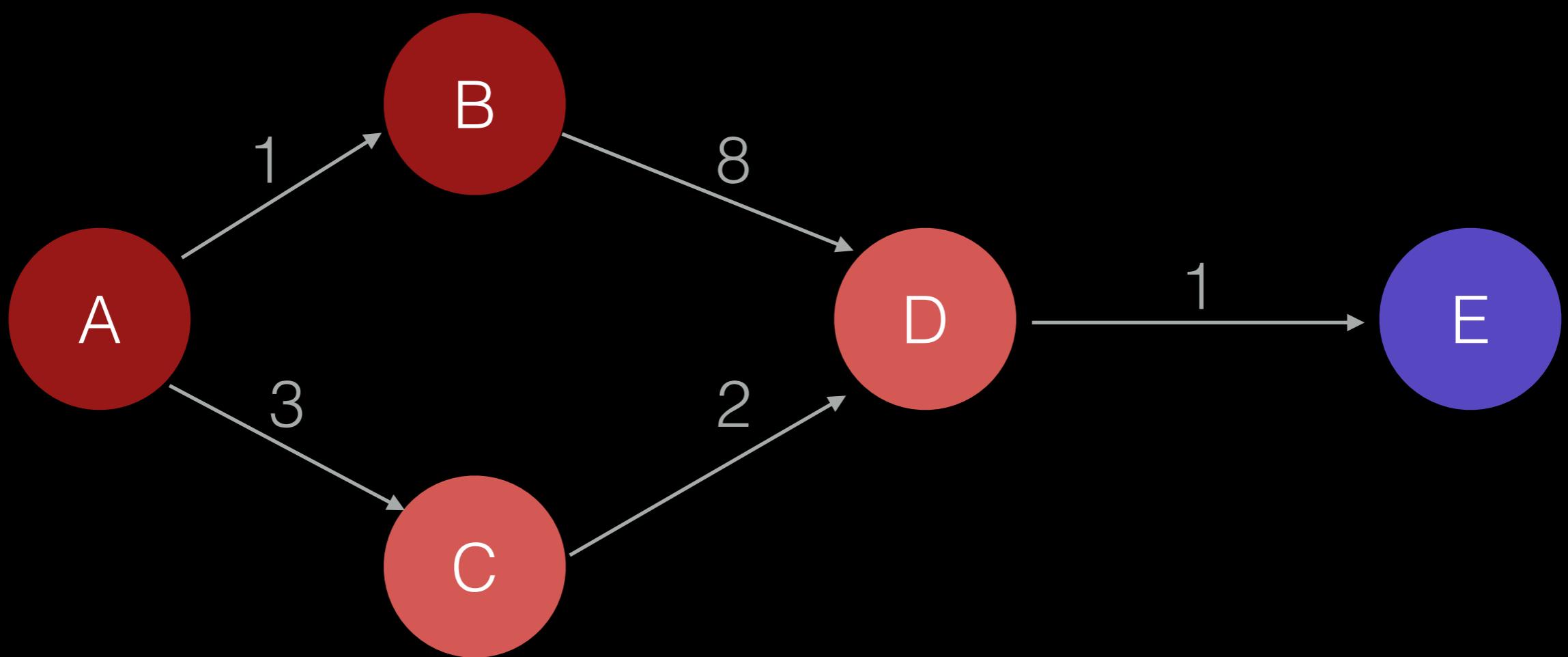
Most problems
use algorithms



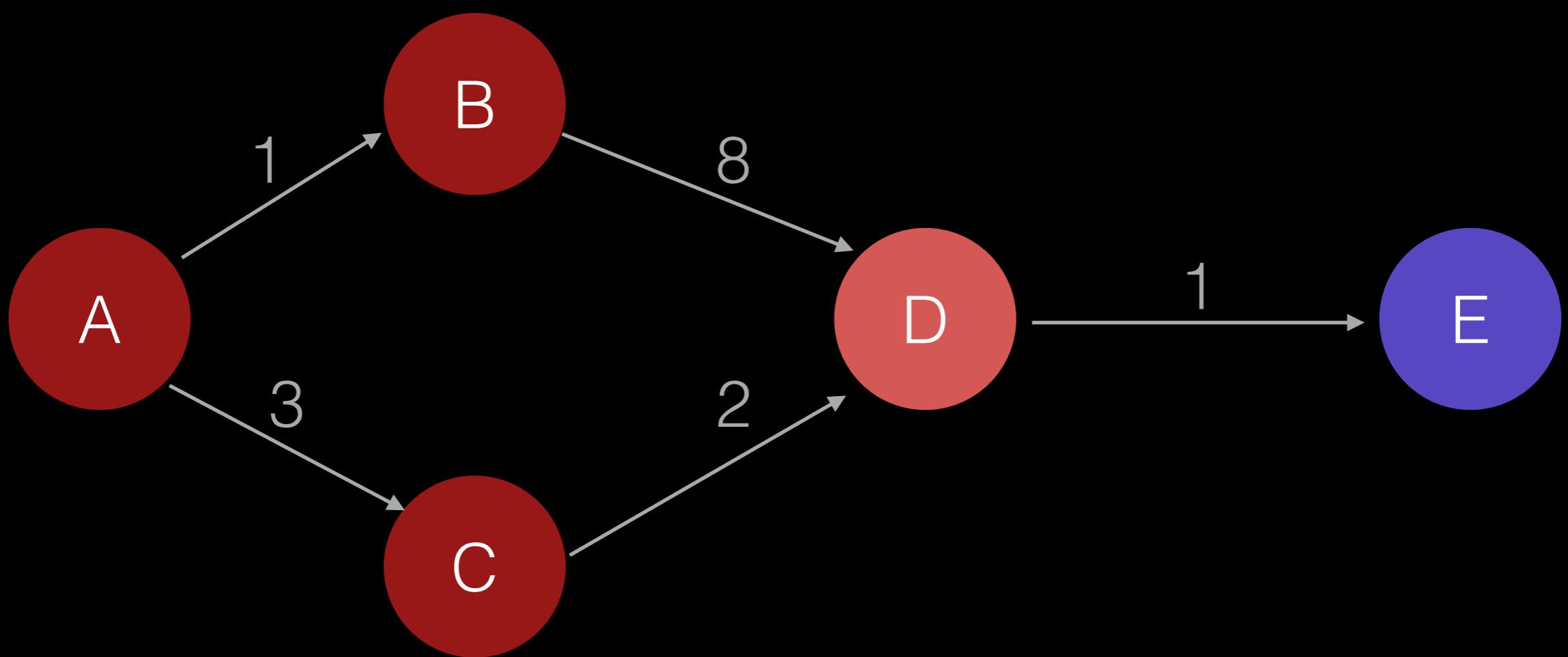
0
A



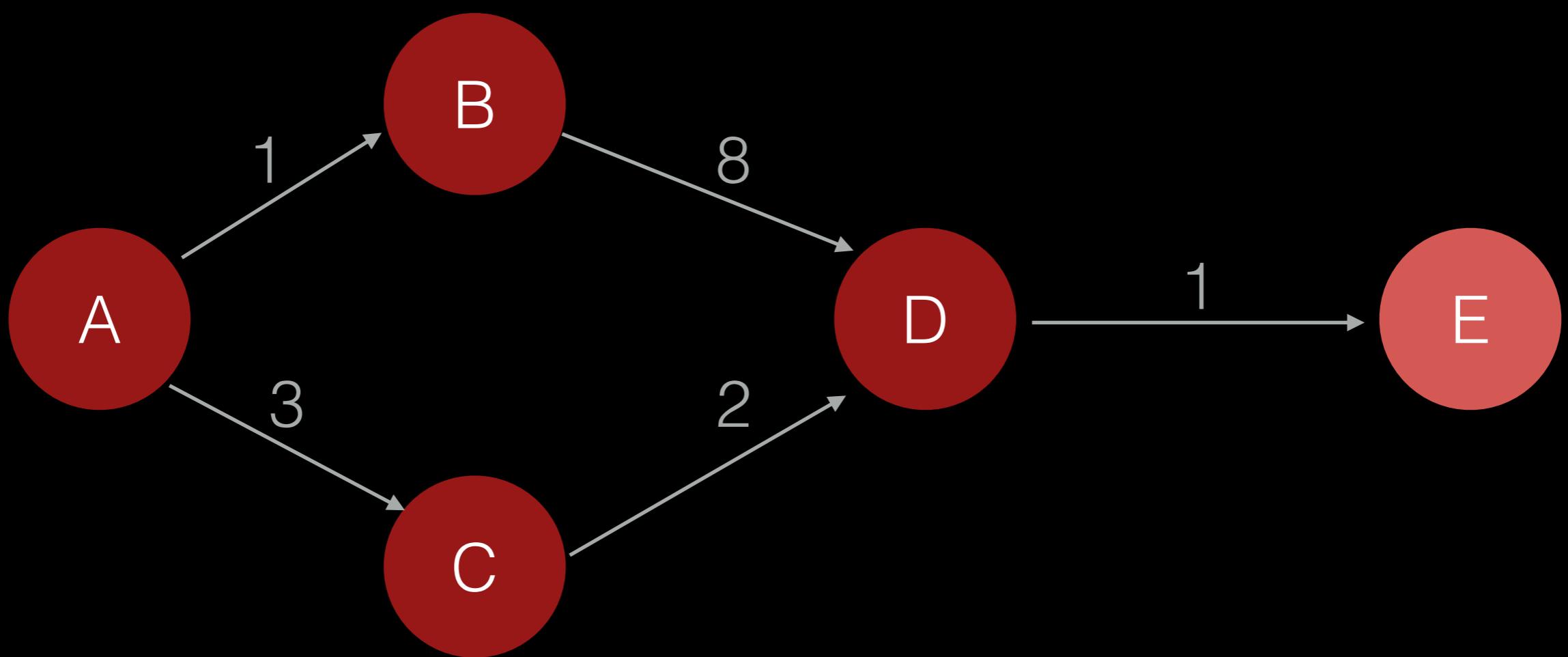
A	0
B	1
C	3



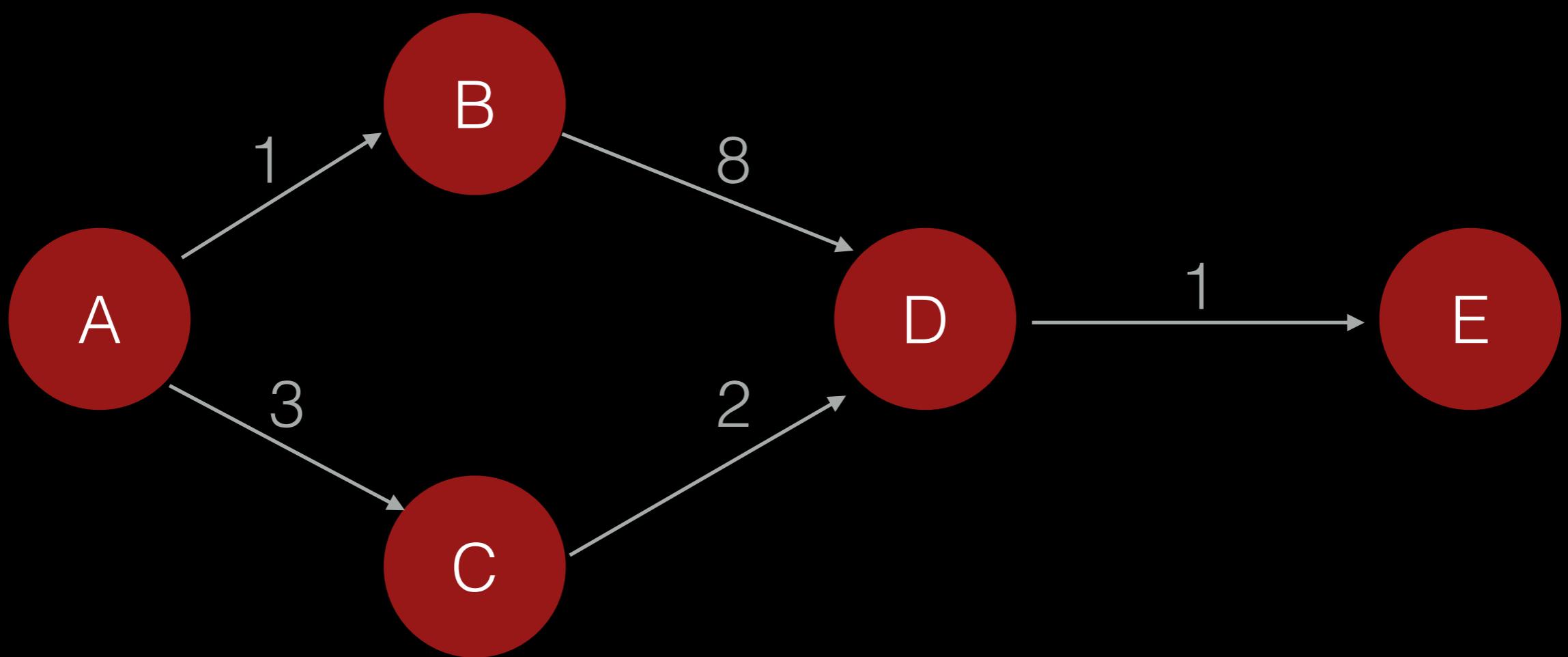
A	0
B	1
C	3
D	9



A	0
B	1
C	3
D	5

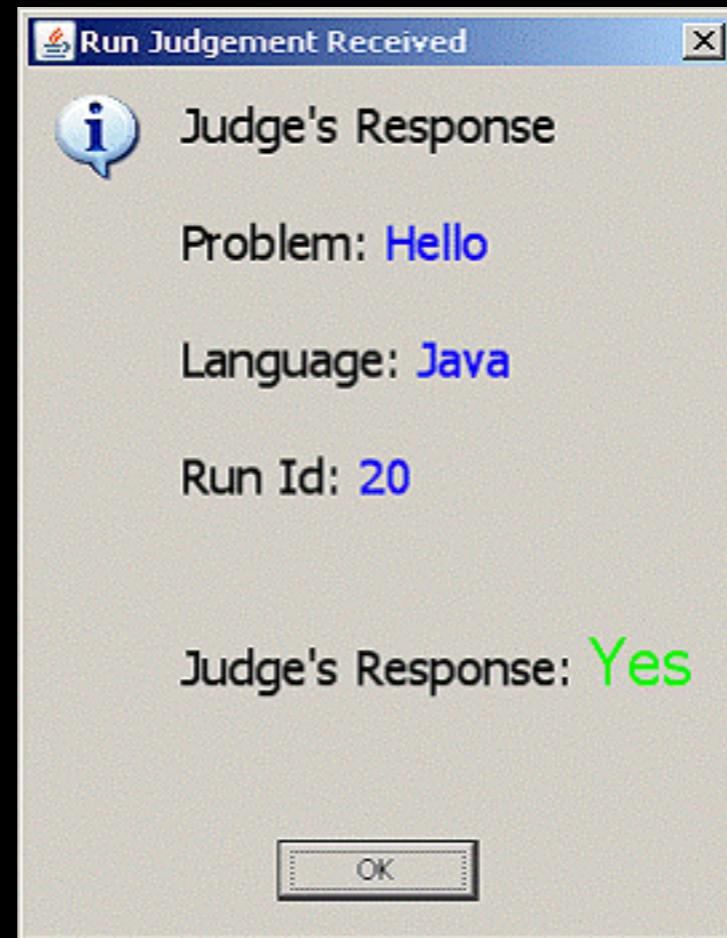


A	0
B	1
C	3
D	5
E	6



A	0
B	1
C	3
D	5
E	6

```
public static void dijkstra(Node root, ArrayList<Node> allNodes)
{ PriorityQueue<Node> q = new PriorityQueue<Node>();
root.distance = 0;
q.add(root);
while(q.size() > 0) {
Node u = q.poll();
for(Node n : u.edges.keySet()) {
if(n.distance == Integer.MAX_VALUE) { // Update the distance to node
n
    q.remove(n);
}
n.distance = Math.min(n.distance, u.distance + u.edges.get(n));
q.add(n); }
} }
```



No - Compilation Error

No - Runtime Exception

No - Time Limit Exceeded

No - Wrong Answer

No - See Contest Staff

No - Compilation Error

No - Runtime Exception

No - Time Limit Exceeded

~10,000,000 operations

No - Wrong Answer

No - See Contest Staff

C

C++

Java

Python*

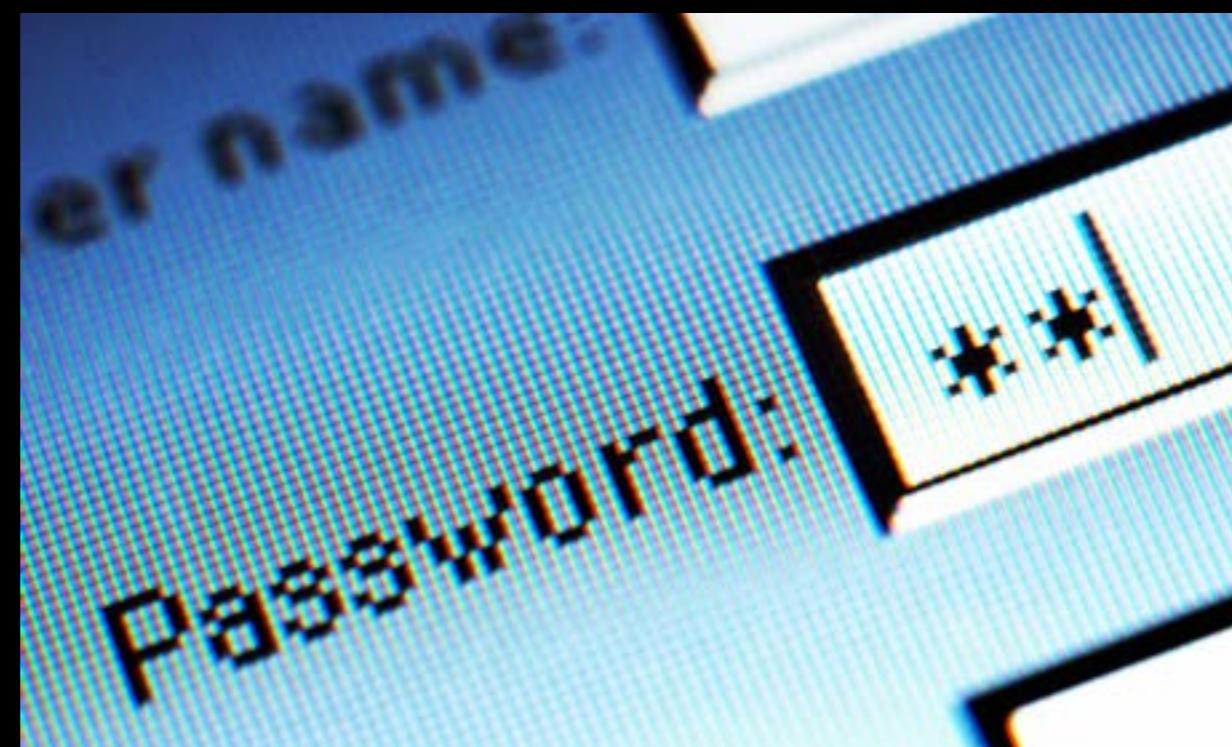
C#*

How to Win

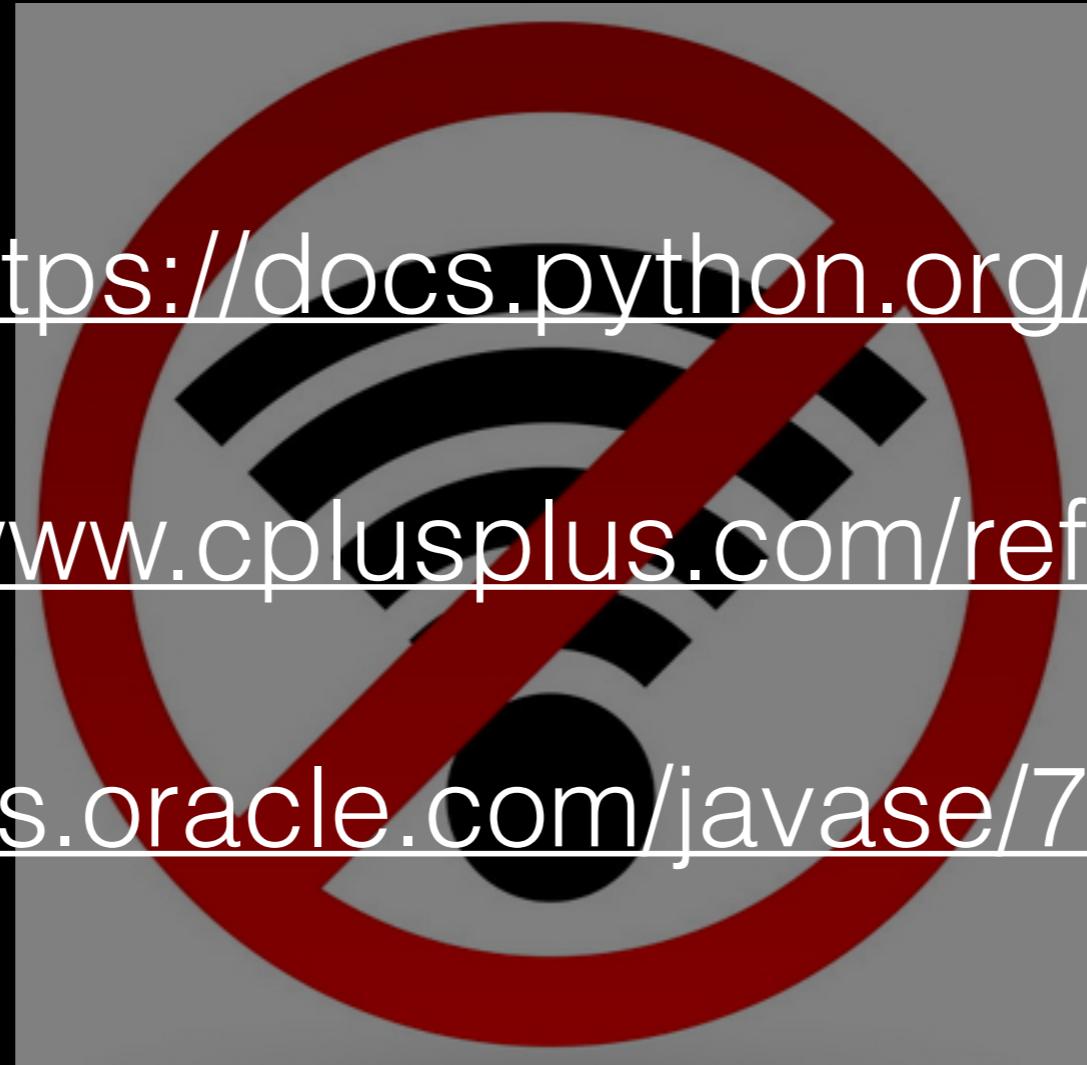
- Solve more problems (1st tiebreaker)
- Solve problems faster (total time 2nd tiebreaker)
 - 20-min penalty for every problem you submit incorrectly, but later get correct
- Submit problems sooner (last correct submission 3rd tiebreaker, likely not used)

More about
the contest

orem ipsum dicitur, ut de
ritur mirabiliter illis
perindea, quae sunt
in aliis.







<https://docs.python.org/2/>

<http://www.cplusplus.com/reference/>

<http://docs.oracle.com/javase/7/docs/api/>

ICPC Regional Competition



- November 5th
- Top 5 teams

ICPC World Finals

- 2011: ~~Sharm el-sheikh, Egypt~~ Orlando, Florida
- 2012: Warsaw, Poland
- 2013: St. Petersburg, Russia
- 2014: Ekaterinburg, Russia
- 2015: Marrakesh, Morocco
- 2016: Phuket, Thailand

ACM-ICPC World Finals

May 20 - 25

2017

Rapid City
South Dakota



home of SDSM&T host ECP, Inc.

world finals

- Schedule
- Activities
- Local Information
- Teams
- World Finals Rules
- Video/Photo Coverage
- World Finals Results
- Past Problems
- Fact Sheet
- Prog. Environment

regionals

- Regional Finder
- Upcoming Regionals
- Regional Results
- Regional Rules
- Getting Involved
- Starting a Regional
- Free ACM Membership

compete

- Preparation
- Policies & Procedures
- FAQs
- The Problems

community

- IBM
- Upsilon Pi Epsilon
- ACM
- Fact Sheet
- ICPC Tools
- History
- Contacts

Schedule of Events

- 9:00am: meet in EEB-105 (**arrive on time!!**)
- 9:30am: move over to CSE 003, practice contest
- 10:30am: full contest begins
- 12:30pm: food arrives
- 3:30pm: contest ends, back to EEB-105
- 4:00pm: end-of-day

Questions?