

# Tompkins

## March 8th, 2014

### Cartoon Mayhem

- **Rules**

- No more than 3 students per team
- There will be 14 programs (including the practice problem) worth 60 points each
  - When a correct program is submitted the team will be awarded 60 points, minus 5 points for each previous incorrect submission
  - Note: The practice problem will not get reduced for multiple submissions.
- Pre-written code is not allowed.
- Team scores will be the total of the teams written scores plus their programming results.

- **Allowed Materials**

- 1 computer
- 1 mouse
- 1 keyboard
- 1 reference book
- 1 printer

# 1. How Far Did I Walk?

- **Input File:** walk.in
- **Output:** Console

## Problem Description

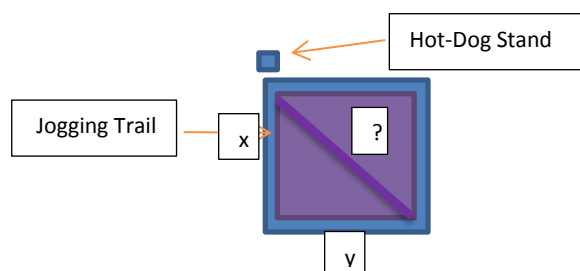
It's a bright sunny day and Garfield plans to have a brisk walk in the park. After reaching the park he goes to the jogging trail and eats a hot-dog from a nearby Hot-Dog stand.

After eating, he starts his jog. He heads south from the Hot-dog stand and travels 'x' meters. Then he takes a 90° turn towards the right. Then he jogs 'y' meters, reaching the next corner. He suddenly feels hungry and cuts across the trail to get back to hot-dog stand.

He wants to know the how far he jogged, but he only knows the distances of the first two legs. So he comes to you to write a program to calculate it for him.

## Notes

1. 'x' or 'y' will be given to you as the input.



## Input Description

There will be 'n' number of inputs. Each set will have an 'x' value and a 'y' value, separated by a space.

## Output Description

The output will be the total distance he traveled. The output will be a double with three decimal places.

## Formulas

Pythagorean Theorem:  $a^2 + b^2 = c^2$

a = first leg 'x'

b = second leg 'y'

c = diagonal

### Sample Input

3

1 5

3 3

4 8

### Sample Output

11.099

10.243

20.944

## 2. Snowball Problem

---

- **Input File:** snowball.in
- **Output:** Console

### Problem Description

Calvin has stored several snowballs in his freezer all winter and spring. Now that it's summer, he's planning to surprise Susie by bombing her with snowballs. However, since he took them out at different times, several of them are beginning to melt in the heat.

Calvin has to determine which snowballs to throw first, based on their varying sizes. He wants you to write a program that will calculate the volume of each snowball and the order he should throw them in.

### Notes

The snowballs will be thrown in order from smallest to largest volume.

The diameters of the snowballs will be within the range  $0 < d < 11$  inches.

The volume of a snowball is calculated by the formula  $V = (4/3) \pi r^3$ .

The numbers in your output should be rounded to the nearest hundredths place.

### Input Description

The first integer represents the number of snowballs that Calvin will throw. Each following line will contain the diameter of a snowball, stored in a double.

### Output Description

The output will be an order list from smallest to largest of the snowballs' volumes. Each volume must be separated by a comma and space.

### Sample Input

```
5
2.34
5.697
10.992
4.0
7.54
```

### Sample Output

```
6.71, 33.51, 96.81, 224.45, 695.39
```

## 3. The Secret Comic

- **Input File:** comic.in
- **Output:** Console

### Problem Description

Mr. Dilbert, a professional comic writer, has written the second part of the very famous Dilbert and Garfield. The publisher told Mr. Dilbert to email him the comic in an encoded language, where the letters are replaced with non-letter characters.

The replacements go as follows:

A: !	J: *	S: 3
B: "	K: +	T: 4
C: #	L: ,	U: 5
D: \$	M: -	V: 6
E: %	N: }	W: 7
F: &	O: /	X: 8
G: '	P: 0	Y: 9
H: (	Q: 1	Z: :
I: )	R: 2	

He comes to you to build him a program to encode his comic.

### Notes

1. The comic is in all caps.
2. There is not punctuation except a period in the comic.
3. In the encoded comic, the spaces and period should remain the same.

### Input Description

The input will have unknown number of lines. You will have to encode them according to the data chart.

### Sample Input

YOU ARE IN TOMPKINS COMPUTER SCIENCE CONTEST.  
HOPE YOU FIND OUR COMPETITION PACKET COMPETITIVE.

### Sample Output

9/5 !2% }} 4/-0+)}3 #/-054%2 3#)%}#% #/}4%34.  
(/0% 9/5 &)}\$ /52 #/-0%4)4/} 0!#+%4 #/-0%4)4)6%.

## 4. The Secret Comic: Part 2

- **Input File:** comic2.in
- **Output:** Console

### Problem Description

Mr. Dilbert's comic was intercepted by Mr. Doonesbury, an all-time rival of Mr. Dilbert. Mr. Doonesbury decoded the comic and made millions of dollars from of it.

The comic publisher calls Dilbert back and tells him to write a new comic and encode it by just putting numbers and adding a '-' after the number value of each letter.

The number value of each letter goes as follows:

A: 1-	J: 10-	S: 19-
B: 2-	K: 11-	T: 20-
C: 3-	L: 12-	U: 21-
D: 4-	M: 13-	V: 22-
E: 5-	N: 14-	W: 23-
F: 6-	O: 15-	X: 24-
G: 7-	P: 16-	Y: 25-
H: 8-	Q: 17-	Z: 26-
I: 9-	R: 18-	

He comes to you to build him a program that encodes this data.

### Notes

1. The comic is in all caps.
2. There is no punctuation except a period in the comic.
3. Each sentence will always have a period.
4. In the encoded comic, the spaces and period should remain the same.
5. The periods and spaces do not need a '-' after them.

### Input Description

The input will have unknown number of lines. You will have to encode them according to the data chart.

### Sample Input

WE HOPE THAT YOU LIKED THE PREVIOUS PART OF THE PROBLEM.  
PLEASE ENJOY CODING THIS ONE.

### Sample Output

23-5- 8-15-16-5- 20-8-1-20- 25-15-21- 12-9-11-5-4- 20-8-5- 16-18-5-22-9-15-21-19- 16-1-18-20- 15-6- 20-8-5- 16-18-15-2-12-5-  
13-.  
16-12-5-1-19-5- 5-14-10-15-25- 3-15-4-9-14-7- 20-8-9-19- 15-14-5-.

## 5. The Secret Comic: The Final Mix-up

- **Input File:** comic3.in
- **Output:** Console

### Problem Description

The publisher of Mr. Dilbert's comic finally received the encoded comic. But now he has a new problem. He needs a fast and easy way to decode it. So he comes to you for a program to decode it.

Each number and dash decodes to the following letters:

1-: A	10-: J	19-: S
2-: B	11-: K	20-: T
3-: C	12-: L	21-: U
4-: D	13-: M	22-: V
5-: E	14-: N	23-: W
6-: F	15-: O	24-: X
7-: G	16-: P	25-: Y
8-: H	17-: Q	26-: Z
9-: I	18-: R	

You need to write a program that decodes this data.

### Notes

1. There is no punctuation except a period in the comic.
2. Each sentence will always have a period.
3. In the decoded comic, the spaces and period should remain the same.
4. The periods and spaces do not have a '-' after them.

### Input Description

The input will have unknown number of lines. You will have to decode them according to the data chart.

### Sample Input

```
23-5- 8-15-16-5- 20-8-1-20- 25-15-21- 12-9-11-5-4- 20-8-5- 16-18-5-22-9-15-21-19- 16-1-18-20- 15-6- 20-8-5- 16-18-15-2-12-5-  
13-.  
16-12-5-1-19-5- 5-14-10-15-25- 3-15-4-9-14-7- 20-8-9-19- 15-14-5-.
```

### Sample Output

```
WE HOPE THAT YOU LIKED THE PREVIOUS PART OF THE PROBLEM.  
PLEASE ENJOY CODING THIS ONE.
```

## 6. Garfield's Nap

---

- **Input File:** nap.in
- **Output:** Console

### Problem Description

Garfield would like to take a very, very, very long nap. However he does not want to be disturbed during his episode of laziness. Therefore he would like you to calculate how long he can sleep without waking up and pick which room is the best to sleep in.

### Input Description

*The first number is the number of rooms you are testing. Every next line is followed with the name of the room, a space, and then a line of letters. In the line, every time the letter z appears, it means Garfield is asleep. Every time another letter appears, it means someone enters the room and Garfield is awakened. Calculate how many times Garfield is woken up in each room and print which room he is disturbed the least.*

### Notes

*There will never be any 2 rooms that he is disturbed the same  
There will always be 11 z's in each room*

### Sample Input

```
3
Bedroom zzzzzuzzzzozzz
Livingroom zzzzozzzzkzzzc
Bathroom zkzpzzzzzszzrzz
```

### Sample Output

```
Bedroom
```



# 7. Garfield's Cannonball

- **Input File:** cannon.in
- **Output:** Console

## Problem Description

The Comic War Z has begun. Garfield, the Allies' leading physicist comes to you make a program which can calculate the distance a cannonball can travel, the time it takes to travel and the maximum height it reaches and the time it takes to reach the ground, when launched from a location.

## Notes

$$t_h = \frac{v_0 \sin(\theta)}{g}$$

*Time to reach max*

$$t_t = 2t_h$$

*Total time in flight*

$$h = v_0 \sin(\theta) t_h - .5g(t_h)^2$$

*Max height*

$$x = v_0 \cos(\theta) t_t$$

*Total distance covered*

## Input Description

There will be unknown number of input sets. Each set will include two lines of data. The first line will be the word Velocity followed by its value. The second line will be the word angle followed by its value.

## Output Description

The output needs to contain the distance a cannonball can travel horizontally; the time it takes to travel to its target and the maximum height it reaches. These outputs must be rounded to the nearest hundredths place.

The output for each data set should be formatted as follows:

Distance: **value**

Time: **value**

Height: **value**

## Assumptions

- $g = 9.8$
- The angle is always greater than 0 and less than 90

### **Sample Input**

Velocity 100

Angle 89

Velocity 986

Angle 45

### **Sample Output**

Distance: 35.61

Time: 20.41

Height: 510.05

Distance: 99203.67

Time: 142.29

Height: 24800.92

## 8. Garfield's Amazing Game

---

- **Input File:** game.in
- **Output:** Console

### Problem Description

Garfield is designing a console game for Arlene. He needs to design mazes for his game. As he is lazy to count the shortest route to the destination, he comes to you to make a program that helps him find the shortest route to complete the maze.

### Input Description

The first line on the input will contain the number of sets to process. In each set, the first line will have the dimensions of the maze, separated by an 'x' (**rowsxcolumns**). Following the dimensions will be the maze.

### Output Description

The output will have the shortest number of steps to complete the maze.

### Assumptions

1. S will always start at location (0,0)
2. X will always be at the bottom-right corner of the maze
3. The maze will always be solvable

### Sample Input

```
2
7x5
S----
WW--W
WW-WW
---WW
--WWW
-----
WWWWX
3x3
S--
-WW
--X
```

### Sample Output

```
12
4
```

## 9. What's That Word, Garfield!

---

- **Input Files:** word.in, wordlist.txt
- **Output:** Console

### Problem Description

Garfield is really bored with his endless vacation. Thus, he plans making a game called Word Scramble on the computer as a Console application. He goes to John to build him a program that gives him a scrambled word and then he needs to type in the actual word.

But, to make his vacation time more interesting, he comes to you to build him a program that finds a solution for a specific scrambled word.

### Input Description

You will be given a file with unscrambled word named "wordlist.txt". This file will be used as the dictionary for your program.

The next file will contain an unknown number of scrambled words. This file will be named "word.in"

### Output Description

The program will print off the unscrambled words in the order they were read.

### Sample Input

aahlp  
aabcillsy  
achkl  
diknr  
eeefrz  
achiilorst  
afinnt

### Sample Output

alpha  
basically  
chalk  
drink  
freeze  
historical  
infant

# 10. Where's My Comic Strip

---

- **Input File:** strip.in
- **Output:** Console

## Problem Description

Doonesbury has written a detective comic strip for Daily Bugle. But as the wind blew in Central Park, the comic strip flew away horizontally. Thus, he needs to walk on the pavement to find his comic strip. Later, he remembers this incident in life and thinks about and coding this incident into a console application.

## *Rules for Calculating the Distance, Time and Speed*

1. Doonesbury starts from the tall tree and always starts towards the left.
  - "T" denotes tall tree
  - "-" denotes an single step
  - "t" denotes trash
  - "C" denotes comic strip
2. There will only be one "T" and "C".
3. Every single step ("-") counts as a single meter and a single second.
4. Once he goes to the end of the given path, he travels back to the tall tree and then goes to the other side.
5. Once he finds the comic strip, he stops adding distance and time.
6. If he finds trash, it takes 3 seconds for him to throw it away.
  - a. After this point this location is treated as a '-', due to it no longer has trash
  - b. After the trash is cleaned he continues on his way
7.  $Speed = \frac{Distance}{Time}$
8. The speed is rounded to the nearest tenth.

## Input / Output Description

The first line will tell you the number of sets to follow. The lines following will have the whole path that Doonesbury has to follow. The output needs to show the total distance traveled, time spent collecting and speed for his comic collection.

You output must fix be in the following format for each data set:

Distance: **value** meters

Time: **value** seconds

Speed: **value** meters per second

### Sample Input

2

---t-t--t--T-----C-t--

ttttttttTttttCtt

### Sample Output

Distance: 27 meters

Time: 33 seconds

Speed: .8 meters per second

Distance: 19 meters

Time: 41 seconds

Speed: .5 meters per second

# 11. Did I Make It In?

---

- **Input File:** track.in
- **Output:** Console

## Problem Description

It is track season at Super Comics School of Comical Education and Coach Dilbert is looking for amazing talents that can win the State Track Meet. There are several leading comic kings coming to watch the try-outs, so Coach Dilbert wants to make it fancy. He types down all run times and comes to you to write a program that prints whether a student did excellent, above average, average or poor, and then prints the top five of the try-outs.

### Notes

1. *If the time is less than or equal 10 seconds: Excellent*
2. *If the time is less than or equal 15 seconds: Above Average*
3. *If the time is less than or equal 17 seconds: Average*
4. *If the time is greater than 17 seconds: Poor*
5. *If there are two or more players with the same time, then the rank will list all the runners with that time in alphabetical order.*
6. *There will always be enough runners to fill all 5 ranks.*

## Input Description

There will be unknown number of lines, each containing the athlete name and the timing (in seconds) separated by a space.

## Output Description

The output will have a list of all players with the description of their timing (excellent, above average, average or poor) in the order in which it was read. Then there will be a separate list of the top five athletes. This list will be titled "TOP-Five".

## Sample Input

```
Jane 10
Todd 18
Tedd 10
Bob 4
Jim 6
Sam 14
Tina 3
Jimbo 22
Steve 17
Mark 14
```

## Sample Output

Jane Excellent

Todd Poor

Tedd Excellent

Bob Excellent

Jim Excellent

Sam Above Average

Tina Excellent

Jimbo Poor

Steve Average

Mark Above Average

TOP-Five

Tina

Bob

Jim

Jane

Tedd



# 12. Date Conversion

---

- **Input File:** date.in
- **Output:** Console

## Problem Description:

Michael wants to find out the number of days that have passed since he was born to the date of several of his achievements. Given a list of dates in the format of mm/dd/yyyy, print out the number of days that have passed since Michael's birthday which is March 8, 1998. The day count starts with his birthday as day one and includes the day to which he wants to count.

## Notes

*You have to account for leap years which have 29 days in February instead of the usual 28. Since the input will never be 2100 or higher, you can assume any year divisible by 4 is a leap year.*

## Input Description

The first line will contain a single integer, n, which represents the number of dates to compute.

The next 'n' lines will have a 2-digit integer representing the month, then a forward slash(/), then a 2-digit integer representing the day, then another forward slash(/), and finally a 4-digit integer representing the year.

All dates will be later than March 8, 1998(03/08/1998) and none will be later than December 31,2099(12/31/2099).

## Sample Input

```
3
03/09/1998
12/22/2009
02/29/2012
```

## Sample Output

```
2
4308
5107
```

# 13. Garfield's Amazing Game Part2

---

- **Input File:** game2.in
- **Output:** Console

## Problem Description

Garfield has come with a new type of maze for his console game. The maze has 2 keys, 2 doors, a start location and an end location. He comes to you to make a program that will tell if his mazes are solvable.

## Symbols

- W – Wall
- S – Starting location
- X – Ending location
- A – Door
- B – Door
- a – Key to door A
- b – Key to door B

## Input Description

The first line on the input will contain the number of sets to process. In each set, the first line will have the dimensions of the maze, separated by an 'x' (**rowsxcolumns**). Following the dimensions will be the maze.

## Output Description

The output will display whether each maze is solvable or not.

## Assumptions

1. S will always start at location (0,0)
2. X will always be at the bottom-right corner of the maze
3. The maze will have 2 doors and 2 keys
4. Key a only opens door A
5. Key b only opens door B
6. You cannot travel through a door unless you have its key

### Sample Input

3

7x5

S---a

WW--W

WWAww

b--ww

WBwww

-----

WWWwX

3x3

SAb

-ww

aBX

3x3

S-a

Aww

BbX

### Sample Output

true

true

false