

William P. Clements

High School

Computer Science Invitational

Novice Division

November 2013

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Team Number: | | | School: | |
| **Problem Checklist**  (Turn in at the end of contest) | | | | |
| **Problem Number** | **Name** | **Tally of Incorrect Submissions (Tiebreaker)** | | **Score** |
| **2 Points** | **1** | Hello, World! |  | |  |
| **2** | Parity Party |  | |  |
| **3** | GCD and LCM |  | |  |
| **4** | Wilbur the Pig |  | |  |
| **5** | Charlottery |  | |  |
| **5 Points** | **6** | Inequality |  | |  |
| **7** | Spacing |  | |  |
| **8** | Factorial Fun |  | |  |
| **9** | Geersh Cone |  | |  |
| **10** | SwoleMaster 2000 |  | |  |
| **9 Points** | **11** | Yang-Mills Theory |  | |  |
| **12** | Farmer Will Billy |  | |  |
| **13** | Collision |  | |  |
| **14** | Where’s Waldo? |  | |  |
| **15** | Lavin the Intern Elf |  | |  |
|  | **Total:** | | | |  |

1. **Hello, World!**

**Program Name: Saluton.java Input File: None**

**Input**

None.

**Output**

Display code that would display code for a “Hello World!” program.

**Example Output to Screen**

public class saluton {

public static void main(String[] args) {

System.out.println("public class saluton {");

System.out.println("public static void main(String[] args){");

System.out.println("System.out.println(\"Hello, World!\");");

System.out.println("}");

System.out.println("}");

}

}

1. **Parity Party**

**Program Name: Parity.java Input File: parity.dat**

CileyMyrus, a famous singer, wants to hold a parity party at her mansion. She can only do so, however, if the sum of the two numbers sent to her by WikeMiLLMadeIt is even.

**Input**

The first line of input is an integer n that represents the number of datasets to follow. Each successive line will contain two integers separated by a space.

**Output**

For each line, if the sum of the two numbers is even, print “We like parity!” Otherwise, print “We can stop :(”.

**Assumption**

* The given integers and their resulting sum will not exceed Integer.MAX\_VALUE.

**Example Input File**

3

12 22

32 35

81 14

**Example Output to Screen**

We like parity!

We can stop :(

We can stop :(

1. **GCD and LCM**

**Program Name: GCD.java Input File: gcd.dat**

Mr. Boyd, a famous coach of Mathletes and athletes alike, is teaching his students Number Sense tricks. He has trouble remembering a certain trick – how to determine the product of the GCD (Greatest Common Denominator) and LCM (Least Common Multiple) of two numbers. Help him recall this Number Sense trick.

**Input**

The first line of input is an integer n that represents the number of datasets to follow. Each successive line will contain two integers separated by a space.

**Output**

For each line, output the product of the GCD and LCM of the two numbers.

**Example Input File**

4

6 14

8 12

5 6

3 5

**Example Output to Screen**

84

96

30

15

1. **Wilbur the Pig**

**Program Name: Pig.java Input File: pig.dat**

Wilbur the Pig was eating when he realized that he’s getting too fat. He decided to keep a schedule of his eating habits. Help Wilbur avoid slaughter by evaluating his weight.

**Input**

The first line of input will provide Wilbur’s initial weight. The second line of input will provide the number of days recorded on his schedule. Each of the following lines will consist of one of two actions taken on a day – EAT or STARVE – and the number of pounds he gains or loses, respectively, on that day.

**Output**

Display Wilbur’s final weight in the format “Wilbur weighs \* pounds.” Replace the ‘\*’ with his final weight.

**Assumption**

* All numbers given are positive integers and will not exceed Integer.MAX\_VALUE.

**Example Input File**

200

12

EAT 6

STARVE 3

EAT 4

EAT 15

STARVE 9

EAT 1

STARVE 6

EAT 23

EAT 4

EAT 123

STARVE 56

EAT 9

**Example Output to Screen**

Wilbur weighs 311 pounds.

1. **Charlottery**

**Program Name: Charlottery.java Input File: charlottery.dat**

Charlotte the Spider is hosting a series of lotteries! Wilbur the Pig wants to participate, but he doesn’t know how to calculate his chances of winning. Help Wilbur evaluate the probability that his tickets win each lottery.

**Input**

The first line of input will provide the number of lotteries. Each of the following lines gives the number of tickets Wilbur bought and the number of total tickets purchased in the lottery, separated by a space. Assume that all numbers are positive integers.

**Output**

Print the probability that Wilbur wins the lottery, expressed as a percentage and rounded to the nearest percent.

**Example Input File**

10

1 2

5 6

2 10

10 50

6 8

9 27

7 84

13 91

2 3

9 15

**Example Output to Screen**

50%

83%

20%

20%

75%

33%

8%

14%

67%

60%

1. **Inequality**

**Program Name: Inequality.java Input File: inequality.dat**

Some Dude needs help with his Algebra 1 homework, in which he must determine whether a point satisfies a given inequality. However, Some Dude does not comprehend the concept of inequalities, so he treats them as if they were regular equations instead. Because Some Dude is too lazy to do his homework, he wants you to do it for him. Being the awesome programmer (and absolutely wonderful friend) that you are, you decide to automate the whole process for him.

**Input**

The first line of input denotes the number of datasets. Each dataset consists of two lines. The first line denotes the inequality to use, in the standard form Ax + By = C, where A, B, and C are integers. The second line has the coordinate to be checked, in the form x y, where x and y are integers, separated by a space.

**Output**

Print out “true” if the point satisfies the equation, or “false” if it does not.

**Example Input File**

2

5x + 6y = 50

3 4

4x + 12y = 8

5 -1

**Example Output to Screen**

false

true

1. **Spacing**

**Program Name: Spacing.java Input File: spacing.dat**

Charlotte was writing words on her web when she noticed that her letters were spaced unevenly. Help Charlotte fix her words.

**Input**

The first line of input will provide the number of words to follow. The following lines contain a series of letters separated by spaces.

**Output**

Display each word that Charlotte wrote with the characters strung together.

**Example Input File**

11

S o M E

P I G

T E r R I F ic

W i l b u r

t h e

P i G

DE L I C i o us

B A c o N

C H A R LOTTE

W A S

H E RE

**Example Output to Screen**

SoME

PIG

TErRIFic

Wilbur

the

PiG

DELICious

BAcoN

CHARLOTTE

WAS

HERE

**8. Fabulous Fantastic Fierce Factorial Fun**

**Program Name: Factorial.java Input File: factorial.dat**

Sampson wants to generate huge numbers in order to decide how many hamburgers he wants to eat. Write a program that will help him do so.

**Input**

The first line of input will provide the number of datasets to follow. Each dataset will contain one single digit positive number.

**Output**

Output the sum of the factorials of all numbers less than or equal to the given number.

**Assumptions**

* A factorial is defined as a product of all the numbers less than or equal to the number. For example, the factorial of 5 is 5\*4\*3\*2\*1.
* The output value will not exceed Integer.MAX\_VALUE.

**Example Input File**

2

5

4

**Example Output to Screen**

153

33

**9. Geersh Cone**

**Program Name: Geersh.java Input File: geersh.dat**

The great Mr. G. D. Shvar wants to buy as many coconuts as possible. Help him find the store with the most coconuts. He can only carry a certain number of coconuts.

**Input**

The first line will provide the number of datasets to follow. The first line of each dataset will be the maximum number of coconuts Mr. Shvar can carry. The next line will contain a number n representing the number of stores that carry coconuts. The next n lines will each consist of the name of the store (one word) and the integer number of coconuts available at that store, separated by a space.

**Output**

Output the store that sells the greatest number of coconuts that he can carry. If there is a tie, choose the one that appears earliest on the list.

**Assumptions**

* There will always be a store he can buy coconuts from.
* Mr. Shvar will only buy coconuts from one store.
* Mr. Shvar will purchase either all or none of the coconuts from a store.
* All numbers given will be positive and will not exceed Integer.MAX\_VALUE.

**Example Input File**

2

15

6

Krager 4

Rondalls 10

HUB 6

Walblues 16

Walbazaar 14

Costca 12

10

4

McDonulds 10

Burgerz 10

Frieys 6

Clemens 16

**Example Output to Screen**

Walbazaar

McDonulds

**10. SwoleMaster 2000**

**Program Name: Swolemaster.java Input File: swolemaster.dat**

Lavin Chellaram, the swolest person in the known universe, has a list of girls on his cell phone. When on his swole tour of Eastern Canada, he finds himself bored and wonders what the statistics of the numbers on his phones are. Although Lavin is the smartest programmer in the known universe, he is too tired from playing DotA2 (in which he is literally better than all of Alliance, Na’Vi, and IG put together). Write a program to help Lavin figure out the statistics of the numbers on his cell phone.

**Input**

The first line will give you the number of lists that Lavin has. Each following line will consist of a list of integers, each separated by a space. These integers are the numbers in Lavin’s list.

**Output**

Output the largest number, smallest number, and difference between the two numbers. Separate groups with a blank line. The last line should read, after a blank line, “I am the SwoleMaster 2000.”

**Assumptions**

* All numbers given are positive and will not exceed Integer.MAX\_VALUE
* Each line will contain at least one and no more than 100 elements.

**Example Input File**

3

2013920493 911 88888888 9146134 1234567890

333 928394039 12345 28 1 2 123

1200394839 2038475943 2038403938 2031830394 29384728

**Example Output to Screen**

Largest Number – 2013920493

Smallest Number – 911

Difference – 2013919582

Largest Number – 928394039

Smallest Number – 1

Difference – 928394038

Largest Number – 2038475943

Smallest Number – 29384728

Difference – 2009091215

I am the SwoleMaster 2000.

**11. Yang-Mills Theory**

**Program Name: Mills.java Input File: mills.dat**

 d = \frac{v^2 \sin(2 \theta)}{g} The distance traveled by a projectile, given velocity and launch angle θ, can be calculated by the following formula:

**Input**

The first line will provide the number of datasets to follow, each consisting of two lines. The first line shows the initial velocity, the second line gives the angle for θ in degrees. Velocity and θ are denoted by v and t, respectively.

**Output**

Display the distance traveled rounded to 6 decimal places.

**Assumptions**

* At William Perry Clements High School, g = 9.79197.
* Launch angle will be between 0 and 90 degrees, inclusive.

**Example Input File**

2

v = 123.23

t = 49

v = 22

t = 13.5

**Example Output to Screen**

1535.732581

22.439959

**12. Farmer Will Billy**

**Program Name: Farmer.java Input File: farmer.dat**

Farmer Will Billy is planning his crops for next year’s harvest. Help Will determine which crops sell the best and grow the fastest by writing a program to calculate each crops’ benefits.

**Input**

The first line of input will consist of the number of crops, at least three.

The following lines each contain a number of elements, separated by spaces:

* Crop name, which will be one word.
* Buying price, in dollars.
* Selling price, in dollars.
* Harvest time, in days.

**Output**

Display profit per day for each crop in the order they appear, rounded to the nearest cent. Profit is the selling price minus the buying price.

**Example Input File**

5

Wilberry 6.00 45.50 100

Moneytree 100.00 1000.00 365

Corn 1.00 2.00 3

Bacon 6.25 18.00 10

Rock 0.01 0.02 1000000

**Example Output to Screen**

Wilberry: $0.40/day

Moneytree: $2.47/day

Corn: $0.33/day

Bacon: $1.18/day

Rock: $0.00/day

**13. Collision**

**Program Name: Collision.java Input File: collision.dat**

Some Dude has gone to Geometry now! And yet again, he is unable to do basic geometry problems involving intersection of geometric shapes, namely circles. It’s up to you to save the day!

**Input**

The first line of input denotes the number of datasets to follow. Each dataset is composed of two lines. The first line contains 3 integer parameters for the first circle, x, y, and r, representing the x-coordinate of the center of the circle, the y-coordinate of the center, and the radius of the circle respectively. The second line contains the parameters for the second circle, ordered identically.

**Output**

Print out “true” if the circles are intersecting, or “false” if it they do not. Consider tangency to be intersecting. If one circle is entirely inside the other, consider them as intersecting.

**Example Input File**2

4 5 8

2 13 1

5 6 3

4 2 1

**Example Output to Screen**

true

false

**14. Where’s Waldo?**

**Program Name: Waldo.java Input File: waldo.dat**

Waldo has learned how to code and has started hiding himself in Strings! Nobody has been able to find him yet because he had suddenly gained the power to replace certain letters in his name. He can only change one letter at a time, and has an unfortunate habit of leaving very obvious clues as to which letter he has changed. See if you can find Waldo.

**Input**

The first line consists of an integer, n, representing the number of Strings to follow. Each String will have three elements, each separated by a space. The first element will be a single character representing which letter in Waldo’s name (‘W’, ‘a’, ‘l’, ‘d’, or ‘o’) is being replaced. The second element will also be a single character representing the replacement letter. The third element will be a string of characters containing Waldo with the replaced letter somewhere.

**Output**

The index at which Waldo’s name occurs in each string, counting the first letter of the third element as the beginning.

**Assumptions**

* Waldo’s name with the replaced letter will always appear exactly once in the third element.
* The third element will not contain any spaces.

**Example Input File**

3

o k OnedoesnotsimplyWaldkintoMordor

W n ringringringringring...Banaldophone

l d DopenguinsWaddoorWalko?

**Example Output to Screen**

16

25

10

**15. Lavin the Intern Elf**

**Program Name: Elf.java Input File: elf.dat**

For his summer internship, Lavin has decided to travel to the North Pole and help Santa Claus with his tax audit. The Internal Revenue Service is very skeptical of Santa’s claims that he has spent large amounts of money on Christmas presents during the holiday season. To prove these claims, Lavin must sort through Santa’s receipts and make a list of how much Santa actually spent during the Christmas season.

**Input**

The first line of input will contain the types of gifts that Santa has given, n. The next n lines will contain the types of gifts in the format “Name of Gift – Price of Gift”. After an empty line, the number of gifts that Santa has given, x, will be displayed. The next x lines will contain the name of the gift that Santa has given.

**Output**

Output the total cost of the gifts that Santa has given.

**Assumptions**

* The accuracy of the price of each toy will not go past the second decimal place.
* Total money spent will not exceed Integer.MAX\_VALUE.
* The number of toys that Santa has purchased will not exceed 10000.

**Example Input File**

13

Tickle Me Elmo - 100.00

Xbox 360 – 399.99

iPhone 4S – 199.99

Barbie Doll – 19.99

Computer – 1000.00

Nike Sneakers – 50.99

Espresso Machine – 150.50

Amazon Kindle – 100.00

Finding Nemo – 15.99

Toy Story 3 – 9.99

BluRay Player – 99.99

Starbucks Gift Card – 9.99

Colored Jeans – 19.99

28

Computer

Tickle Me Elmo

BluRay Player

Tickle Me Elmo

Starbucks Gift Card

Computer

Tickle Me Elmo

Barbie Doll

Starbucks Gift Card

Espresso Machine

Tickle Me Elmo

Starbucks Gift Card

Espresso Machine

Finding Nemo

Barbie Doll

Starbucks Gift Card

Amazon Kindle

Tickle Me Elmo

Espresso Machine

Finding Nemo

Espresso Machine

Tickle Me Elmo

Espresso Machine

Computer

Starbucks Gift Card

Finding Nemo

Finding Nemo

Tickle Me Elmo

**Example Output to Screen**

Santa spent $4806.38 on Christmas Gifts.