

24.1 LAB: Mad Lib - loops



This section has been set as optional by your instructor.

Mad Libs are activities that have a person provide various words, which are then used to complete a short story in unexpected (and hopefully funny) ways.

Write a program that takes a string and an integer as input, and outputs a sentence using the input values as shown in the example below. The program repeats until the input string is **quit** and disregards the integer input that follows.

Ex: If the input is:

```
apples 5
shoes 2
quit 0
```

the output is:

```
Eating 5 apples a day keeps you happy and healthy.
Eating 2 shoes a day keeps you happy and healthy.
```

539740.3879454.qx3zqy7

LAB
ACTIVITY

24.1.1: LAB: Mad Lib - loops

0 / 10



main.cpp

[Load default template...](#)

```
1 #include <iostream>
2 #include <string>
3 using namespace std;
4
5 int main() {
6
7     /* Type your code here. */
8
9     return 0;
10 }
11
```

©zyBooks 01/31/24 18:13 1939727
Rob Daglio
MDCCOP2335Spring2024

Develop mode**Submit mode**

Run your program as often as you'd like, before submitting for grading. Below, type any needed input values in the first box, then click **Run program** and observe the program's output in the second box.

Enter program input (optional)

If your code requires input values, provide them here.

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

Run program

Input (from above)

**main.cpp**
(Your program)

Output

Program output displayed here

Coding trail of your work

[What is this?](#)

History of your effort will appear here once you begin working on this zyLab.

24.2 LAB: Checker for integer string



This section has been set as optional by your instructor.

Forms often allow a user to enter an integer. Write a program that takes in a string representing an integer as input, and outputs Yes if every character is a digit 0-9.

Ex: If the input is:

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

1995

the output is:

Yes

Ex: If the input is:

or

the output is:

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

Hint: Use a loop and the isdigit() function (don't forget to include the ctype library).

539740.3879454.qx3zqy7

LAB
ACTIVITY

24.2.1: LAB: Checker for integer string

0 / 10



main.cpp

Load default template...

```
1 #include <iostream>
2 #include <string>
3 using namespace std;
4
5 int main() {
6     string userString;
7
8     cin >> userString;
9
10    /* Type your code here. */
11
12    return 0;
13 }
14
```

Develop mode

Submit mode

Run your program as often as you'd like, before submitting for grading. Below, type any needed input values in the first box, then click **Run program** and observe the program's output in the second box.

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

Enter program input (optional)

Run program

Input (from above)

main.cpp
(Your program)

Output

Program output displayed here

Coding trail of your work [What is this?](#)

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

History of your effort will appear here once you begin working on this zyLab.

24.3 LAB: Password modifier



This section has been set as optional by your instructor.

Many user-created passwords are simple and easy to guess. Write a program that takes a simple password and makes it stronger by replacing characters using the key below, and by appending "!" to the end of the input string.

- i becomes 1
- a becomes @
- m becomes M
- B becomes 8
- s becomes \$

Ex: If the input is:

mypassword

the output is:

Myp@\$\$word!

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

539740.3879454.qx3zqy7

LAB
ACTIVITY

24.3.1: LAB: Password modifier

0 / 10



main.cpp

[Load default template...](#)

1 `#include <iostream>`

```
2 #include <string>
3 using namespace std;
4
5 int main() {
6
7     /* Type your code here. */
8
9     return 0;
10 }
```

©zyBooks 01/31/24 18:13 1939727
Rob Daglio
MDCCOP2335Spring2024

Develop mode**Submit mode**

Run your program as often as you'd like, before submitting for grading. Below, type any needed input values in the first box, then click **Run program** and observe the program's output in the second box.

Enter program input (optional)

If your code requires input values, provide them here.

Run program

Input (from above)

**main.cpp**
(Your program)

Output

Program output displayed here

Coding trail of your work [What is this?](#)

History of your effort will appear here once you begin working on this zyLab.

©zyBooks 01/31/24 18:13 1939727
Rob Daglio
MDCCOP2335Spring2024

24.4 LAB: Count input length without spaces, periods, exclamation points, or commas



This section has been set as optional by your instructor.

Given a line of text as input, output the number of characters excluding spaces, periods, exclamation points, or commas.

Ex: If the input is:

Listen, Mr. Jones, calm down.

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

the output is:

21

Note: Account for all characters that aren't spaces, periods, exclamation points, or commas (Ex: "r", "2", "?").

539740.3879454.qx3zqy7

LAB
ACTIVITY

24.4.1: LAB: Count input length without spaces, periods, exclamation points, or commas

0 /

10



main.cpp

Load default template...

```
1 #include <iostream>
2 #include <string>
3 using namespace std;
4
5 int main() {
6     string userText;
7     // Add more variables as needed
8
9     getline(cin, userText); // Gets entire line, including spaces.
10
11     /* Type your code here. */
12
13     return 0;
14 }
15
```

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

Develop mode

Submit mode

Run your program as often as you'd like, before submitting for grading. Below, type any needed input values in the first box, then click **Run program** and observe the program's output in the second box.

Enter program input (optional)

If your code requires input values, provide them here.

Run program

Input (from above)



main.cpp
(Your program)



Output

Program output displayed here

©zyBooks 01/31/24 18:13 1939727
Rob Daglio
MDCCOP2335Spring2024

Coding trail of your work [What is this?](#)

History of your effort will appear here once you begin working on this zyLab.

24.5 LAB: Countdown until matching digits



This section has been set as optional by your instructor.

Write a program that takes in an integer in the range 11-99 (inclusive) as input. The output of the program is a countdown starting from the input integer until an integer where both digits are identical. End with a newline.

Ex: If the input is:

93

the output is:

93 92 91 90 89 88

©zyBooks 01/31/24 18:13 1939727
Rob Daglio
MDCCOP2335Spring2024

Ex: If the input is:

11

the output is:

11

Ex: If the input is:

9

or any value not between 11 and 99 (inclusive), the output is:

Input must be 11–99

@zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

For coding simplicity, follow each output number by a space, even the last one.

Use a while loop. Compare the digits; do not write a large if-else for all possible same-digit numbers (11, 22, 33, ..., 99), as that approach would be cumbersome for larger ranges.

539740.3879454.qx3zqy7

LAB
ACTIVITY

24.5.1: LAB: Countdown until matching digits

0 / 10



main.cpp

Load default template...

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5
6     /* Type your code here. */
7
8     return 0;
9 }
10
```

Develop mode

Submit mode

Run your program as often as you'd like, before submitting for grading. Below, type any needed input values in the first box, then click **Run program** and observe the program's output in the second box.

@zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

Enter program input (optional)

If your code requires input values, provide them here.

[Run program](#)

Input (from above)

**main.cpp**
(Your program)

Output

Program output displayed here

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

Coding trail of your work

[What is this?](#)

History of your effort will appear here once you begin working on this zyLab.

24.6 LAB: Palindrome



This section has been set as optional by your instructor.

A palindrome is a word or a phrase that is the same when read both forward and backward. Examples are: "bob," "sees," or "never odd or even" (ignoring spaces). Write a program whose input is a word or phrase, and that outputs whether the input is a palindrome.

Ex: If the input is:

`bob`

the output is:

`palindrome: bob`

Ex: If the input is:

`bobby`

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

the output is:

`not a palindrome: bobby`

Hint: Start by just handling single-word input, and submit for grading. Once passing single-word test cases, extend the program to handle phrases. If the input is a phrase, remove or ignore spaces.

539740.3879454.qx3zqy7



main.cpp

1 Loading latest submission...|

©zyBooks 01/31/24 18:13 1939727
Rob Daglio
MDCCOP2335Spring2024

Develop mode

Submit mode

Run your program as often as you'd like, before submitting for grading. Below, type any needed input values in the first box, then click **Run program** and observe the program's output in the second box.

Enter program input (optional)

If your code requires input values, provide them here.

Run program

Input (from above)



main.cpp
(Your program)



Output

Program output displayed here

©zyBooks 01/31/24 18:13 1939727
Rob Daglio
MDCCOP2335Spring2024

Coding trail of your work [What is this?](#)



Retrieving signature

24.7 LAB: Brute force equation solver



This section has been set as optional by your instructor.

Numerous engineering and scientific applications require finding solutions to a set of equations. Ex: $8x + 7y = 38$ and $3x - 5y = -1$ have a solution $x = 3, y = 2$. Given integer coefficients of two linear equations with variables x and y , use brute force to find an integer solution for x and y in the range -10 to 10 .

Ex: If the input is:

```
8 7 38
3 -5 -1
```

the output is:

```
x = 3, y = 2
```

Use this brute force approach:

```
For every value of x from -10 to 10
    For every value of y from -10 to 10
        Check if the current x and y satisfy both equations. If so,
        output the solution, and finish.
```

Ex: If no solution is found, output:

```
There is no solution
```

Assume the two input equations have no more than one solution.

Note: Elegant mathematical techniques exist to solve such linear equations. However, for other kinds of equations or situations, brute force can be handy.

539740.3879454.qx3zqy7

LAB
ACTIVITY

24.7.1: LAB: Brute force equation solver

©zyBooks 01/31/24 18:13 1939727
Rob Daglio 0 / 10
MDCCOP2335Spring2024

main.cpp

[Load default template...](#)

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
```

```
5
6  /* Type your code here. */
7
8  return 0;
9 }
10
```

©zyBooks 01/31/24 18:13 1939727
Rob Daglio
MDCCOP2335Spring2024

Develop mode**Submit mode**

Run your program as often as you'd like, before submitting for grading. Below, type any needed input values in the first box, then click **Run program** and observe the program's output in the second box.

Enter program input (optional)

If your code requires input values, provide them here.

Run program

Input (from above)



main.cpp
(Your program)



Output

Program output displayed here

Coding trail of your work [What is this?](#)

History of your effort will appear here once you begin working on this zyLab.

24.8 LAB: Count multiples

©zyBooks 01/31/24 18:13 1939727
Rob Daglio
MDCCOP2335Spring2024



This section has been set as optional by your instructor.

Write a program that takes three integers as input: low, high, and x. The program then outputs the number of multiples of x between low and high exclusive.

Ex: If the input is:

1 10 2

the output is:

Multiples of 2: 2,4,6,8,

@zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

For coding simplicity, follow each output number by a comma, even the **last one**.

Hint: Use the % operator to determine if a number is a multiple of x. Use a for loop to test each number between low and high.

539740.3879454.qx3zqy7

**LAB
ACTIVITY**

24.8.1: LAB: Count multiples

0 / 10



main.cpp

1 Loading latest submission...



Develop mode

Submit mode

Run your program as often as you'd like, before submitting for grading. Below, type any needed input values in the first box, then click **Run program** and observe the program's output in the second box.

@zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

Enter program input (optional)

If your code requires input values, provide them here.

Run program

Input (from above)

**main.cpp**
(Your program)

Output

Program output displayed here

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

Coding trail of your work [What is this?](#)

Retrieving signature

24.9 LAB: Find largest number



This section has been set as optional by your instructor.

Write a program that repeatedly reads in integers until a negative integer is read. The program also keeps track of the largest integer that has been read so far and outputs the largest integer at the end.

Ex: If the input is:

`2 77 17 4 -1`

the output is:

`Largest integer: 77`

Assume a user will enter at least one non-negative integer.

539740.3879454.qx3zqy7

**LAB
ACTIVITY**

24.9.1: LAB: Find largest number

0 / 10



©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

main.cpp**Load default template...**

```

1 #include <iostream>
2 using namespace std;
3
4 int main() {
5
6     /* Type your code here. */

```

```
7  
8 return 0;  
9 }
```

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

Develop mode**Submit mode**

Run your program as often as you'd like, before submitting for grading. Below, type any needed input values in the first box, then click **Run program** and observe the program's output in the second box.

Enter program input (optional)

If your code requires input values, provide them here.

Run program

Input (from above)

**main.cpp**
(Your program)

Output

Program output displayed here

Coding trail of your work [What is this?](#)

History of your effort will appear here once you begin working on this zyLab.

24.10 LAB: Hailstone sequence

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024



This section has been set as optional by your instructor.

Given a positive integer n , the following rules will always create a sequence that ends with 1, called the hailstone sequence:

- If n is even, divide it by two

- If n is odd, multiply it by 3 and add 1 (i.e. $3n + 1$)
- Continue until n is 1

Write a program that reads an integer as input and prints the hailstone sequence starting with the integer entered. Format the output so that five integers, each separated by a tab character (`\t`), are printed per line. End the output with a tab character.

The output format can be achieved as follows:

```
cout << n << "\t";
```

Ex: If the input is:

25

the output is:

25	76	38	19	58
29	88	44	22	11
34	17	52	26	13
40	20	10	5	16
8	4	2	1	

539740.3879454.qx3zqy7

©zyBooks 01/31/24 18:13 1939727
Rob Daglio
MDCCOP2335Spring2024

LAB
ACTIVITY

24.10.1: LAB: Hailstone sequence

0 / 10



main.cpp

Load default template...

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5
6     /* Type your code here. */
7
8     return 0;
9 }
```

©zyBooks 01/31/24 18:13 1939727
Rob Daglio
MDCCOP2335Spring2024

Develop mode

Submit mode

Run your program as often as you'd like, before submitting for grading. Below, type any needed input values in the first

box, then click **Run program** and observe the program's output in the second box.

Enter program input (optional)

If your code requires input values, provide them here.

Run program

Input (from above)



©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024
main.cpp
(Your program)



Output

Program output displayed here

Coding trail of your work [What is this?](#)

History of your effort will appear here once you begin working on this zyLab.

24.11 LAB: Months until payoff



This section has been set as optional by your instructor.

Write a program that reads a loan amount, payment amount, and interest rate as inputs and outputs the number of payments required until the loan is paid. Interest is added to current balance *before* a payment is applied. Ex: If current balance is \$100.00 and the interest rate is 0.02, the new balance is \$102.00 before a payment is applied. All values are doubles.

Ex: If the input is:

1000.0 50.0 0.03

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

the output is:

31 payments

Ex: If the input is:

50.0 100.0 0.02

the output is:

1 payment

539740.3879454.qx3zqy7

LAB
ACTIVITY

24.11.1: LAB: Months until payoff

©zyBooks 01/31/24 18:13 1939727

Rob Daglio 0 / 10
MDCCOP2335Spring2024



main.cpp

Load default template...

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5
6     /* Type your code here. */
7
8     return 0;
9 }
```

Develop mode

Submit mode

Run your program as often as you'd like, before submitting for grading. Below, type any needed input values in the first box, then click **Run program** and observe the program's output in the second box.

Enter program input (optional)

If your code requires input values, provide them here.

Run program

Input (from above)



main.cpp

(Your program)



Output

Program output displayed here

Coding trail of your work [What is this?](#)

History of your effort will appear here once you begin working on this zyLab.

24.12 LAB: Matching strings

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024



This section has been set as optional by your instructor.

Write a program that compares two strings given as input. Output the number of characters that match in each string position. The output should use the correct verb (match vs matches) according to the character count.

Ex: If the input is:

crash crush

the output is:

4 characters match

Ex: If the input is:

cat catnip

the output is:

3 characters match

Ex: If the input is:

mall saw

the output is:

1 character matches

Ex: If the input is:

apple orange

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

the output is:

0 characters match

539740.3879454.qx3zqy7

LAB
ACTIVITY

24.12.1: LAB: Matching strings

0 / 10



©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

[Load default template...](#)

main.cpp

```
1 #include <iostream>
2 #include <string>
3 using namespace std;
4
5 int main() {
6     /* Type your code here. */
7
8
9     return 0;
10 }
```

Develop mode

Submit mode

Run your program as often as you'd like, before submitting for grading. Below, type any needed input values in the first box, then click **Run program** and observe the program's output in the second box.

Enter program input (optional)

If your code requires input values, provide them here.

Run program

Input (from above)



main.cpp

(Your program)



Output

Program output displayed here

Coding trail of your work

[What is this?](#)

History of your effort will appear here once you begin working on this zyLab.

24.13 LAB: Draw right-justified triangle

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024



This section has been set as optional by your instructor.

Write a program to draw a right-justified triangle given the height as input. The first row has one asterisk (*) and increases by one for each row. Each asterisk is followed by a blank space and each row ends with a newline.

Ex: If the input is:

3

the output is:

```
  *
 * *
* * *
```

539740.3879454.qx3zqy7

LAB
ACTIVITY

24.13.1: LAB: Draw right-justified triangle

0 / 10



main.cpp

Load default template...

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     ... /* Type your code here. */
6
7     return 0;
8 }
9 }
```

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

Develop mode**Submit mode**

Run your program as often as you'd like, before submitting for grading. Below, type any needed input values in the first box, then click **Run program** and observe the program's output in the second box.

Enter program input (optional)

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

If your code requires input values, provide them here.

Run program

Input (from above)

**main.cpp**
(Your program)

Output

Program output displayed here

Coding trail of your work

[What is this?](#)

History of your effort will appear here once you begin working on this zyLab.

24.14 LAB: Draw upside down triangle



This section has been set as optional by your instructor.

Write a program that outputs a right triangle of asterisks given the height as input. Each line ends with a blank space.

Ex: If the input is:

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

3

the output is:

```
* * *  
* *
```

*

539740.3879454.qx3zqy7

LAB
ACTIVITY

24.14.1: LAB: Draw upside down triangle

0 / 10



main.cpp

©zyBooks 01/31/24 18:13 1939727
Load default template...

Rob Daglio

MDCCOP2335Spring2024

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5
6     /* Type your code here. */
7
8     return 0;
9 }
```

Develop mode

Submit mode

Run your program as often as you'd like, before submitting for grading. Below, type any needed input values in the first box, then click **Run program** and observe the program's output in the second box.

Enter program input (optional)

If your code requires input values, provide them here.

Run program

Input (from above)

main.cpp
(Your program)

Output

Program output displayed here

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

Coding trail of your work

[What is this?](#)

History of your effort will appear here once you begin working on this zyLab.

24.15 LAB*: Program: Drawing a half arrow

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024



This section has been set as optional by your instructor.

Program Specifications Write a program that outputs a downwards facing arrow composed of a rectangle and a right triangle. Arrow dimensions are defined by user specified arrow base height, arrow base width, and arrow head width.

Note: this program is designed for *incremental development*. Complete each step and submit for grading before starting the next step. Only a portion of tests pass after each step but confirm progress.

Step 1 (3 pts). Input the arrow base height (int) and width (int). Draw a rectangle using asterisks (height x width). Hint: use a nested loop in which the inner loop draws one row of *s, and the outer loop iterates a number of times equal to the height. Submit for grading to confirm two tests pass.

Ex: If input is:

6 4

Sample output is:

```
*****
*****
*****
*****
*****
*****
```

Step 2 (3 pts). Input the arrow head width and draw a right triangle. Hint: use a nested loop. Submit for grading to confirm four tests pass.

©zyBooks 01/31/24 18:13 1939727

Rob Daglio

MDCCOP2335Spring2024

Ex: If input is:

4 3 4

Sample output is:


```

***
***
***
***
****
***
**
*

```

©zyBooks 01/31/24 18:13 1939727
Rob Daglio
MDCCOP2335Spring2024

Step 3 (4 pts). Modify the program to only accept an arrow head width that is larger than the arrow base width. Use a loop to continue inputting the arrow head width until the value is larger than the arrow base width. Submit for grading to confirm all tests pass.

Ex: If input is:

4 3 3 2 4

Sample output is:

```

***
***
***
***
****
***
**
*

```

539740.3879454.qx3zqy7

LAB
ACTIVITY

24.15.1: LAB*: Program: Drawing a half arrow

0 / 10



main.cpp

Load default template...

```

1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     /* Type your code here. */
6
7     return 0;
8 }
9

```

©zyBooks 01/31/24 18:13 1939727
Rob Daglio
MDCCOP2335Spring2024

Develop mode**Submit mode**

Run your program as often as you'd like, before submitting for grading. Below, type any needed input values in the first box, then click **Run program** and observe the program's output in the second box.

Enter program input (optional)

If your code requires input values, provide them here.

Run program

Input (from above)


main.cpp
 (Your program)


Output

Program output displayed here

Coding trail of your work [What is this?](#)

History of your effort will appear here once you begin working on this zyLab.

24.16 LAB*: Program: Rock paper scissors



This section has been set as optional by your instructor.

Program Specifications Write a program to play an automated game of Rock, Paper, Scissors. Two players make one of three hand signals at the same time. Hand signals represent a rock, a piece of paper, or a pair of scissors. Each combination results in a win for one of the players. Rock crushes scissors, paper covers rock, and scissors cut paper. A tie occurs if both players make the same signal. Use a random number generator of 0, 1, or 2 to represent the three signals.

Note: this program is designed for *incremental development*. Complete each step and submit for grading before starting the next step. Only a portion of tests pass after each step but confirm progress.

Step 0. Read starter template and *do not* change the provided code. Integer constants are defined for ROCK, PAPER, and SCISSORS. A Random object is created and a seed is read from input and passed to the Random object. This supports automated testing and creates predictable results that would otherwise be random.

Step 1 (2 pts). Read two player names from input (string). Read number of rounds from input. Continue reading number of rounds if value is below one and provide an error message. Output player names and number of rounds. Submit for grading to confirm 2 tests pass.
Ex: If input is:

```
3 Anna Bert -3 -4 4
```

Sample output is:

```
Rounds must be > 0
Rounds must be > 0
Anna vs Bert for 4 rounds
```

Step 2 (2 pts). Generate random values (0 - 2) for player 1 followed by player 2 by calling `rand() % 3`. Continue to generate random values for both players until both values do not match. Output "Tie" when the values match. Submit for grading to confirm 3 tests pass.
Ex: If input is:

```
10 Anna Bert 1
```

Sample output is:

```
Anna vs Bert for 1 rounds
Tie
Tie
```

Step 3 (3 pts). Identify winner for this round and output a message. Rock crushes scissors, scissors cut paper, and paper covers rock. Submit for grading to confirm 6 tests pass.
Ex: If input is:

```
39 Anna Bert 1
```

Sample output is:

```
Anna vs Bert for 1 rounds
Tie
```

Tie
Bert wins with scissors

Step 4 (3 pts). Add a loop to repeat steps 2 and 3 for the number of rounds. Output total wins for each player after all rounds are complete. Submit for grading to confirm all tests pass.

Ex: If input is:

©zyBooks 01/31/24 18:13 1939727

82 Anna Bert 3

Rob Daglio
MDCCOP2335Spring2024

Sample output is:

Anna vs Bert for 3 rounds
Anna wins with paper
Tie
Anna wins with rock
Anna wins with paper
Anna wins 3 and Bert wins 0

539740.3879454.qx3zqy7

LAB
ACTIVITY

24.16.1: LAB*: Program: Rock paper scissors

0 / 10



main.cpp

Load default template...

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     const int ROCK = 0;
6     const int PAPER = 1;
7     const int SCISSORS = 2;
8
9     int seed;
10    cin >> seed;
11    srand(seed);
12
13    /* Type your code here. */
14
15    return 0;
```

©zyBooks 01/31/24 18:13 1939727

Rob Daglio
MDCCOP2335Spring2024

Develop mode

Submit mode

Run your program as often as you'd like, before submitting for grading. Below, type any needed input values in the first box, then click **Run program** and observe the program's output in the second box.

Enter program input (optional)

If your code requires input values, provide them here.

Run program

Input (from above)



main.cpp

(Your program)



Output

Program output displayed here

Coding trail of your work

[What is this?](#)

History of your effort will appear here once you begin working on this zyLab.

©zyBooks 01/31/24 18:13 1939727
Rob Daglio
MDCCOP2335Spring2024