

23.1 LAB: Remove gray from RGB



This section has been set as optional by your instructor.

©zyBooks 01/31/24 18:09 1939727

Rob Daglio
MDCCOP2335Spring2024

Summary: Given integer values for red, green, and blue, subtract the gray from each value.

Computers represent color by combining the sub-colors red, green, and blue (rgb). Each sub-color's value can range from 0 to 255. Thus (255, 0, 0) is bright red, (130, 0, 130) is a medium purple, (0, 0, 0) is black, (255, 255, 255) is white, and (40, 40, 40) is a dark gray. (130, 50, 130) is a faded purple, due to the (50, 50, 50) gray part. (In other words, equal amounts of red, green, blue yield gray).

Given values for red, green, and blue, remove the gray part.

Ex: If the input is:

130 50 130

the output is:

80 0 80

Find the smallest value, and then subtract it from all three values, thus removing the gray.

Note: [This page](#) converts rgb values into colors.

539740.3879454.qx3zqy7

LAB ACTIVITY

23.1.1: LAB: Remove gray from RGB

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
```

©zyBooks 01/31/24 18:09 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.

©zyBooks 01/31/24 18:09 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

Coding trail of your work

[What is this?](#)

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

23.2 LAB: Seasons



This section has been set as optional by your instructor.

Write a program that takes a date as input and outputs the date's season in the northern hemisphere. The input is a string to represent the month and an int to represent the day. Note: End with a newline.

©zyBooks 01/31/24 18:09 1939727

Ex: If the input is:

Rob Daglio

MDCCOP2335Spring2024

April 11

the output is:

Spring

In addition, check if the string and int are valid (an actual month and day).

Ex: If the input is:

```
Blue 65
```

the output is:

```
Invalid
```

@zyBooks 01/31/24 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024

The dates for each season in the northern hemisphere are:

Spring: March 20 - June 20

Summer: June 21 - September 21

Autumn: September 22 - December 20

Winter: December 21 - March 19

539740.3879454.qx3zqy7

LAB
ACTIVITY

23.2.1: LAB: Seasons

0 / 10



main.cpp

Load default template...

```
1 #include <iostream>
2 #include <string>
3 using namespace std;
4
5 int main() {
6     string inputMonth;
7     int inputDay;
8
9     cin >> inputMonth;
10    cin >> inputDay;
11
12    /* Type your code here. */
13
14    return 0;
15 }
```

Develop mode

Submit mode

@zyBooks 01/31/24 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024

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 18:09 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.

23.3 LAB: Max of 2



This section has been set as optional by your instructor.

Write a program that takes in two integers as inputs and outputs the largest value. If the two integers are the same, output the integers' value.

Ex: If the input is:

4 2

the output is:

Max of 4 and 2 is 4

539740.3879454.qx3zqy7

©zyBooks 01/31/24 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024

0 / 10

**LAB ACTIVITY**

23.3.1: LAB: Max of 2

```
1 #include <iostream>
2 using namespace std;
```

main.cpp[Load default template...](#)

```
1 using namespace std;
2
3 int main() {
4     int val1;
5     int val2;
6     int max;
7
8     /* Type your code here */
9
10    return 0;
11
12 }
```

©zyBooks 01/31/24 18:09 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 hereCoding 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 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024

23.4 LAB: Max of 3



This section has been set as optional by your instructor.

Write a program that takes in three integers as inputs and outputs the largest value. If the input integers are the same, output the integers' value.

Ex: If the input is:

```
1 2 3
```

the output is:

©zyBooks 01/31/24 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024

```
Max of [1, 2, 3] is 3
```

539740.3879454.qx3zqy7

LAB ACTIVITY

23.4.1: LAB: Max of 3

0 / 10



main.cpp

[Load default template...](#)

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int val1;
6     int val2;
7     int val3;
8     int max;
9
10    /* Type your code here */
11
12    return 0;
13 }
```

[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 18:09 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

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

©zyBooks 01/31/24 18:09 1939727

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

Rob Daglio

MDCCOP2335Spring2024

23.5 LAB: Longest string



This section has been set as optional by your instructor.

Write a program that takes two strings and outputs the longer string, followed by "is longer than", and the shorter string. If the strings have the same length then output the strings followed by "have the same length".

Ex. If the input is:

almond pistachio

the output is:

pistachio is longer than almond

539740.3879454.qx3zqy7

LAB ACTIVITY

23.5.1: LAB: Longest string

0 / 10



main.cpp

Load default template...

©zyBooks 01/31/24 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024

```
1 #include <iostream>
2 #include <string>
3 using namespace std;
4
5 int main() {
6     string str1;
7     string str2;
8     string longest;
9     string shortest;
```

```
10
11  /* Type your code here */
12
13  return 0;
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.

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.

23.6 LAB: Golf scores



This section has been set as optional by your instructor.

©zyBooks 01/31/24 18:09 1939727
Rob Daglio
MDCCOP2335Spring2024

Golf scores record the number of strokes used to get the ball in the hole. The expected number of strokes varies from hole to hole and is called par (possible values: 3, 4, or 5). Each score's name is based on the actual strokes taken compared to par:

- "Eagle": number of strokes is two less than par
- "Birdie": number of strokes is one less than par

- "Par": number of strokes equals par
- "Bogey": number of strokes is one more than par

Given two integers that represent the number of strokes used and par, write a program that prints the appropriate score name. Print "Error" at the end of the output if par or score is not in the expected range.

Ex: If the input is:

©zyBooks 01/31/24 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024

3 4

the output is:

Par 4 in 3 strokes is Birdie

539740.3879454.qx3zqy7

**LAB
ACTIVITY**

23.6.1: LAB: Golf scores

0 / 10



main.cpp

[Load default template...](#)

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

[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 18:09 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.

23.7 LAB: Median of 3



This section has been set as optional by your instructor.

Write a program that takes in three integers and outputs the median value (not the largest or smallest value). End the output with a newline.

Ex: If the input is:

7 1 4

the output is:

4

539740.3879454.qx3zqy7

LAB ACTIVITY

23.7.1: LAB: Median of 3

0 / 10



©zyBooks 01/31/24 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024

[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 18:09 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.

23.8 LAB: Comparing doubles

©zyBooks 01/31/24 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024



This section has been set as optional by your instructor.

Write a program that is given two doubles for comparison and a third double that is a difference threshold, called epsilon. Output one of three phrases:

- output "equal" if the doubles are within 0.001(exclusively) of each other

- output "close enough" if the doubles are within epsilon (exclusively) of each other
- output "not close" if doubles are *not* within epsilon

Note: End each output statement with a newline.

Ex: If the input is:

14.1 14.2 0.2

©zyBooks 01/31/24 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024

the output is:

close enough

Ex: If the input is:

2.1125 2.1132 0.02

the output is:

equal

Ex: If the input is:

5.1 5.0 0.05

the output is:

not close

539740.3879454.qx3zqy7

LAB
ACTIVITY

23.8.1: LAB: Comparing doubles

0 / 10



main.cpp

Load default template...

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

©zyBooks 01/31/24 18:09 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.

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

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

23.9 LAB: Speeding ticket



This section has been set as optional by your instructor.

Write a program that is given two integers representing a speed limit and driving speed in miles per hour (mph) and outputs the traffic ticket amount.

@zyBooks 01/31/24 10:09 1029727

Rob Daglio

MDCCOP2335Spring2024

Driving 10 mph *under* the speed limit (or slower) receives a \$50 ticket. Driving 6 - 20 mph over the speed limit receives a \$75 ticket. Driving 21 - 40 mph over the speed limit receives a \$150 ticket. Driving faster than 40 mph over the speed limit receives a \$300 ticket. Otherwise, no ticket is received.

Ex: If the input is:

35 45

the output is:

75

Ex: If the input is:

35 26

©zyBooks 01/31/24 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024

the output is:

0

539740.3879454.qx3zqy7

LAB ACTIVITY

23.9.1: LAB: Speeding ticket

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.

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 18:09 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.

23.10 LAB: Movie ticket prices



This section has been set as optional by your instructor.

Write a program that takes in a string that holds the values "day" or "night" and an integer that holds a person's age, and outputs a movie ticket price. Movie prices are free for everyone under the age of 4. Daytime prices are \$8 for everyone age 4 or higher. Nighttime prices are \$12 for ages 4 - 16, \$15 for ages 17 - 54 and \$13 for ages 55 and above.

Ex: If the input is:

night 50

the output is:

\$15

Ex: If the input is:

day 15

©zyBooks 01/31/24 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024

the output is:

\$8

Ex: If the input is:

night 3

the output is:

free

539740.3879454.qx3zqy7

LAB ACTIVITY

23.10.1: LAB: Movie ticket prices

©zyBooks 01/31/24 18:09 1939727

Rob Daglio 0 / 10

MDCCOP2335Spring2024



main.cpp

Load default template...

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5
6     /* Your code goes 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**
MDCCOP2335Spring2024
(Your program)

Output

©zyBooks 01/31/24 18:09 1939727

Rob Daglio

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 18:09 1939727

Rob Daglio
MDCCOP2335Spring2024

23.11 LAB: How many digits



This section has been set as optional by your instructor.

Write a program that takes an integer (0 - 9999) as input and outputs the number of digits.

Ex: If the input is:

7493

the output is:

4 digits

Ex: If the input is:

7

the output is:

1 digit

539740.3879454.qx3zqy7

LAB ACTIVITY

23.11.1: LAB: How many digits

0 / 10



main.cpp

©zyBooks 01/31/24 18:09 1939727

Load default template...

MDCCOP2335Spring2024

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5
6     /* Your code goes here */
7
8 }
```

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

©zyBooks 01/31/24 18:09 1939727

Rob Daglio

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

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

23.12 LAB: Phrases and subphrases



This section has been set as optional by your instructor.

©zyBooks 01/31/24 18:09 1939727

Rob Daglio
MDCCOP2335Spring2024

Write a program that reads two phrases on separate lines and outputs one of the four responses:

- 1) Phrase one is found within phrase two
- 2) Phrase two is found within phrase one

3) Both phrases match

4) No matches

Hint: Use the string function `find()`.

Ex: If the input is:

```
fire  
firetruck
```

©zyBooks 01/31/24 18:09 1939727
Rob Daglio
MDCCOP2335Spring2024

the output is:

```
fire is found within firetruck
```

Ex: If the input is:

```
the green grass grows  
green grass
```

the output is:

```
green grass is found within the green grass grows
```

Ex: If the input is:

```
pick a card  
pick a card
```

the output is:

```
Both phrases match
```

Ex: If the input is:

```
apples  
oranges
```

the output is:

©zyBooks 01/31/24 18:09 1939727
Rob Daglio
MDCCOP2335Spring2024

```
No matches
```

539740.3879454.qx3zqy7

**LAB
ACTIVITY**

23.12.1: LAB: Phrases and subphrases

0 / 10



main.cpp

1 Loading latest submission...

©zyBooks 01/31/24 18:09 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?](#)

©zyBooks 01/31/24 18:09 1939727



Retrieving signature

Rob Daglio

MDCCOP2335Spring2024

23.13 LAB: Count odd numbers



This section has been set as optional by your instructor.

Write a program that takes in four positive integers and outputs the number of odd numbers. (Hint: use the modulo operator to determine if a number is odd)

Ex: If the input is:

1 2 3 4

©zyBooks 01/31/24 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024

the output is:

2

Ex: If the input is:

19 19 19 19

the output is:

4

539740.3879454.qx3zqy7

**LAB
ACTIVITY**

23.13.1: LAB: Count odd numbers

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
11
```

©zyBooks 01/31/24 18:09 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)

MD**main.cpp**35Spring2024 → Output

(Your program)

©zyBooks 01/31/24 18:09 1939727

Rob Daglio

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.

23.14 LAB: Alphabetical order (chars)



This section has been set as optional by your instructor.

Write a program that takes in three lowercase characters and outputs the characters in alphabetical order.

Hint: Ordering three characters takes six permutations.

Ex: If the input is:

c a b

©zyBooks 01/31/24 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024

the output is:

a b c

539740.3879454.qx3zqy7

LAB ACTIVITY

23.14.1: LAB: Alphabetical order (chars)



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
11 |
```

©zyBooks 01/31/24 18:09 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 18:09 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.

23.15 LAB*: Program: Income tax form



This section has been set as optional by your instructor.

Program Specifications Write a program to calculate U.S. income tax owed given wages, taxable interest, unemployment compensation, status (single or married) and taxes withheld. Taxpayers are only allowed to use this short form if adjusted gross income (AGI) is less than \$120000. Dollar amounts are displayed as integers with no comma separators. For example, `cout << "Deduction: $" << deduction`

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 (2 pts). Input wages, taxable interest, unemployment compensation, status (1=single and 2=married) and taxes withheld as integers. Calculate and output AGI (wages + interest + unemployment). Output error message if AGI is above \$120000 and the program stops with no additional output. Submit for grading to confirm two tests pass.

Ex: If the input is:

```
20000 23 500 1 400
```

The output is:

```
AGI: $20523
```

Ex: If the input is:

```
120000 23 500 1 400
```

The output is:

```
AGI: $120523  
Error: Income too high to use this form
```

©zyBooks 01/31/24 18:09 1939727
Rob Daglio
MDCCOP2335Spring2024

Step 2 (3 pts). Identify deduction amount based on status: (1) Single=\$12000 or (2) Married=\$24000. Set status to 1 if not input as 1 or 2. Calculate taxable income (AGI - deduction). Set taxable income to zero if negative. Output deduction and taxable income. Submit for grading to confirm five tests pass.
Ex: If the input is:

20000 23 500 1 400

Ex: The additional output is:

```
AGI: $20523
Deduction: $12000
Taxable income: $8523
```

©zyBooks 01/31/24 18:09 1939727

Rob Daglio
MDCCOP2335Spring2024

Step 3 (3 pts). Calculate tax amount based on exemption and taxable income (see tables below). Tax amount should be stored as a double and rounded to the nearest whole number using round(). Submit for grading to confirm eight tests pass.

Ex: If the input is:

20000 23 500 1 400

Ex: The additional output is:

```
AGI: $20523
Deduction: $12000
Taxable income: $8523
Federal tax: $852
```

Income	Tax for Single Filers
\$0 - \$10000	10% of the income
\$10001 - \$40000	\$1000 + 12% of the amount over \$10000
\$40001 - \$85000	\$4600 + 22% of the amount over \$40000
over \$85000	\$14500 + 24% of the amount over \$85000

Income	Tax for Married Filers
\$0 - \$20000	10% of the income
\$20001 - \$80000	\$2000 + 12% of the amount over \$20000
over \$80000	\$9200 + 22% of the amount over \$80000

©zyBooks 01/31/24 18:09 1939727
Rob Daglio
MDCCOP2335Spring2024

Step 4 (2 pts). Calculate amount of tax due (tax - withheld). If the amount due is negative the tax payer receives a refund. Output tax due or tax refund as positive values. Submit for grading to confirm

all tests pass.

Ex: If the input is:

```
80000  0  500  2  12000
```

Ex: The additional output is:

```
AGI: $80500
Deduction: $24000
Taxable income: $56500
Federal tax: $6380
Tax refund: $5620
```

©zyBooks 01/31/24 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024

539740.3879454.qx3zqy7

**LAB
ACTIVITY**

23.15.1: LAB*: Program: Income tax form

0 / 10



main.cpp

[Load default template...](#)

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

©zyBooks 01/31/24 18:09 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 18:09 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.

23.16 LAB*: Program: Grade calculator



This section has been set as optional by your instructor.

Program Specifications Write a program to calculate a course grade given points for homework, quizzes, midterm exam, and final exam. Grades are calculated differently for undergrads, grads and distance learners.

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 (2 pts). Read from input student status (string), homework points, quiz points, midterm exam score, and final exam score (double). Valid student status includes undergrad (UG), grad (G), and distance learner (DL). Calculate each category average using maximum points for homework (800), quizzes (400), midterm exam (150), and final exam (200). Output an error message if student status is not one of the three options. Otherwise, output category averages as a percentage using `cout << "Homework: " << homework << "%" << endl;`. Submit for grading to confirm two tests pass.

©zyBooks 01/31/24 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024

Ex: If the input is:

UG 600.0 300.0 120.0 185.0

The output is:

Homework: 75.0%
Quizzes: 75.0%
Midterm: 80.0%
Final Exam: 92.5%

Ex: If the input is:

TL 600 300 120 180

©zyBooks 01/31/24 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024

The output is:

Error: student status must be UG, G or DL

Step 2 (2 pts). Set any average to 100% if average is above 100%. Submit for grading to confirm three tests pass.

Ex: If the input is:

UG 700.0 300.0 200.0 205.0

The output is:

Homework: 87.5%
Quizzes: 75.0%
Midterm: 100.0%
Final Exam: 100.0%

Step 3 (2 pts). Calculate the course average based on student status using the table below. Output the course average. Submit for grading to confirm five tests pass.

Category	UG	G	DL
Homework	20%	15%	5%
Quizzes	20%	5%	5%
Midterm	30%	35%	40%
Final Exam	30%	45%	50%

©zyBooks 01/31/24 18:09 1939727
Rob Daglio
MDCCOP2335Spring2024

Ex: If the input is:

G	800.0	400.0	100.0	100.0
---	-------	-------	-------	-------

The output is:

```
Homework: 100.0%
Quizzes: 100.0%
Midterm: 66.7%
Final Exam: 50.0%
G average: 65.8%
```

©zyBooks 01/31/24 18:09 1939727
Rob Daglio
MDCCOP2335Spring2024

Step 4 (4 pts). Identify the course letter grade based on the course average using the table below.
Output the course letter grade. Submit for grading to confirm all tests pass.

Average	Grade
at least 90.0	A
at least 80.0 and less than 90.0	B
at least 70.0 and less than 80.0	C
at least 60.0 and less than 70.0	D
less than 60.0	F

Ex: If the input is:

DL	600.0	300.0	120.0	150.0
----	-------	-------	-------	-------

The output is:

```
Homework: 75.0%
Quizzes: 75.0%
Midterm: 80.0%
Final Exam: 75.0%
DL average: 77.0%
Course grade: C
```

©zyBooks 01/31/24 18:09 1939727
Rob Daglio
MDCCOP2335Spring2024

539740.3879454.qx3zqy7

LAB ACTIVITY

23.16.1: LAB*: Program: Grade calculator

0 / 10



main.cpp

[Load default template...](#)

```
1 #include <iostream>
2 #include <iomanip>
3 using namespace std;
4
5 int main() {
6     double HOMEWORK_MAX = 800.0;
7     double QUIZZES_MAX = 400.0;
8     double MIDTERM_MAX = 150.0;
9     double FINAL_MAX = 200.0;
10
11    /* Type your code here. */
12
13    return 0;
14 }
15
```

©zyBooks 01/31/24 18:09 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 18:09 1939727
Rob Daglio
MDCCOP2335Spring2024

23.17 LAB: Phone number string breakdown

Given a string representing a 10-digit phone number, output the area code, prefix, and line number using the format (800) 555-1212.

Ex: If the input is:

```
8005551212
```

the output is:

©zyBooks 01/31/24 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024

```
(800) 555-1212
```

Hint: Use a string operation that returns substrings.

For simplicity, assume all phone numbers are 10-digit. So 18005551212 is not allowed.

539740.3879454.qx3zqy7

**LAB
ACTIVITY**

23.17.1: LAB: Phone number string breakdown

0 / 10



main.cpp

[Load default template...](#)

```
1 #include <iostream>
2 #include <string>
3
4 using namespace std;
5
6 int main() {
7     string phoneNumber;
8
9     cin >> phoneNumber;
10
11    /* Type your code here */
12
13    return 0;
14 }
15
```

[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 18:09 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.

©zyBooks 01/31/24 18:09 1939727

Rob Daglio

MDCCOP2335Spring2024