

Due no later than 9:00pm on Tuesday 9/7.

As you read the following OLI pages and complete the interactive activities, capture the screenshots of the completed activities and replace the respective screenshots in the document.

- Page 7 Arithmetic expressions and simple algorithms
 - Page 8 Integer divisions and related algorithms
 - Page 9 Assignment statements
 - Page 10 Type casting
-
- **When you are ready to submit the assignment, save the document in PDF format and submit the PDF file on Cougar Courses as the proof for your work.**

Page 7 Arithmetic expressions and simple algorithms

Hotspot Arithmetic expression

`first_number * second_number`

How might the above expression affect the value stored in `first_number` and `second_number`?

- ☐ Both `first_number` and `second_number` will take on new values.
- ☒ No change will be imposed to the value stored in `first_number` or `second_number`.
- ☐ Only the variable `first_number` will take on a new value.
- ☐ Only the variable `second_number` will take on a new value.

✓ Correct, participating in an arithmetic operation does not affect the value of the variable on either side of the operator.

In the following space, write an arithmetic expression that would subtract the value stored in `first_number` from the value stored in `second_number`.

`second_number - first_number`

Resubmit

✓ `second_number - first_number`

In the following space, write an arithmetic expression that would divide the value stored in `second_number` by the value stored in `first_number`.

`second_number / first_number`

Resubmit

✓ `second_number / first_number`

LBD evaluate arithmetic expressions

Predict the expected result of the following arithmetic expressions.

$10 - 4 * 2$ evaluates to

✓ Correct! Multiplication has higher precedence over subtraction.

$5 * (10 - 8)$ evaluates to

✓ Correct! The parentheses force the subtraction be evaluated before multiplication.

$2.5 / 0.5 * 2$ evaluates to

✓ Correct! Division and multiplications are of the same precedence. They are left associative, i.e. the left one is performed before the right one.

$2.5 / (0.5 * 2)$ evaluates to

✓ Correct! The parentheses force multiplication be performed before division.

LBD convert math divisions

Convert the expressions into C++ arithmetic expressions

$$\frac{3+5}{4*2}$$

✓ Correct. It's best practice to add spaces between digits and operators so your code is more readable.

$$\frac{5*6}{4+2}$$

✓ Correct. Adding parentheses to the fraction give us an appropriate conversion expression.

$$\frac{2+4}{3/5}$$

✓ Correct. This would give us the correct response.

Checkpoint conversion

According to the program, what would the above speed limit be in kilometers per hour?

✓ Correct! $65 * 1.60934$ is about 104.6

In the program, which line causes the program to pause and wait for user input?

- ☐ Line 6: `int speed_limit;`
- ☐ Line 9: `cout << "What is the speed limit in miles per hour? ";`
- ☒ Line 10: `cin >> speed_limit;`
- ☐ Line 13: `cout << speed_limit << " miles per hour is equivalent to ";`

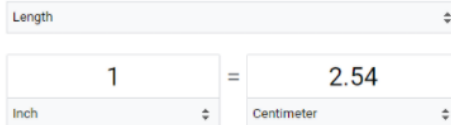
✓

How would the program run differently then the original program? If you are not sure, go ahead run the program in Repl.it.

The program would say, "0 miles per hour is equivalent to a speed limit of 0 kilometers per hour." Then on a new line ask the question, "What is the speed limit in miles per hour?"

Resubmit

✓ Thank you for your response!



[Image Credit](#)

Each inch is equivalent to 2.54 centimeter. Would you be able to modify the above program to convert someone's height in inches into centimeters or meters? Why or why not?

Yes would be able to by storing the value of the inches the user inputs into a variable and taking that value of the variable and multiplying that value by 2.54 in order to convert inches into centimeters or by 0.0254 to convert inches into meters

```
cout << "You are " << inches * 2.54 / 100 << " meters.";
```

Resubmit

Thank you for your response! Your output statement should look something like this.

✓ `cout << "You are " << inches * 2.54 / 100 << " meters.";`

Checkpoint discount & tip

Suppose we have a 15% off coupon and would like to purchase an item that costs \$12.99 originally. According to the program, what is the cost after discount? . How much is the saving?

✓ Correct! $12.99 * (1 - 0.15)$ is rounded to 11.04.

✓ Correct! $12.99 * 0.15$ is rounded to 1.95.

I would feel comfortable writing the necessary output statements to display the appropriate tip amount and total because all I would need to do is ask the user the amount of their original total, ask the percentage of tip they want to tip, take those numbers and find the amount the tip is based of the percentage and original total and then add the tip to the original total to get the new total that includes the tip.

Resubmit

✓ Thanks for sharing.

MR simple problem

Describe a simple problem that you can apply the above 3-step algorithm pattern to solve.

A simple problem that I can add a 3-step algorithm pattern to solve is grocery shopping by getting the groceries prices, add them up and add tax.

Resubmit

✓ Thanks for sharing.

Page 8 Integer divisions and related algorithms

LBD evaluate integer divisions

$21 / 4$ evaluates to

✓ Correct! When both operands are integers, the `/` operator calculates the quotient and ignores the remainder. This is called floor division.

$5 \% 9$ evaluates to

✓ Correct! The modulo operator `%` returns the remainder from the division, which is 5.

$13 / 5$ evaluates to

✓ Correct! The answer is 2 since we are returning the integer quotient, also called the "floor".

$2 * 5 / 3$ evaluates to

✓ Correct.

$11 \% (4 + 1)$ evaluates to

✓ Correct! The remainder of 11 divided by 5 is 1.

$22 \% 6$ evaluates to

✓ Correct! When you divide 22 by 6 you get a remainder of 4

$4 / 5$ evaluates to

✓ Correct! $4 / 5$ evaluates to 0.

$3 \% 9$ evaluates to

✓ Correct! $3 \% 9$ evaluates to 3

$(12 - 2) / 5$ evaluates to

✓ Correct! $10 / 5$ evaluates to the quotient, which is 2.

$(4 + 6) \% 5$ evaluates to

✓ Correct! $10 \% 5$ evaluates to 0 because there is the remainder when 10 is divided by 5.

LBD algorithms with integer divisions

The following code segment has completed steps 1 & 2 of an algorithm that is designed to help a vet convert a puppy's weight in ounces only into pounds and ounces.

```
//Step 1: Declare variables
int weight; //to store the weight of a new born puppy

//Step 2: Collect user input
cout << "How many ounces does the puppy weigh? ";
cin >> weight;
```

Choose the right expressions for the following statement to show the weight in its corresponding pounds and remaining ounces.

cout << "The puppy weighs " << << " lb. " << << " oz.\n";

✓ Correct, this will determine how many pounds the puppy weighs.

✓ Correct, the remaining ounces is the remainder portion of the weight divided by the number of ounces per pound.

```
//Step 1: Declare variables
int tacos; //to store the number of tacos
int guests; //to store the number guests in the party

//Step 2: Collect user input
cout << "How many tacos in the tray? ";
cin >> tacos;
cout << "How many guests in the party? ";
cin >> guests;
```

Given the above code segment. Choose the right expressions for the following statements.

cout << "Each guest should get " << << " tacos.\n";

cout << "After that, there would be " << << " tacos left.\n";

✓ Correct, the quotient of this division will help us know how many tacos each guest should get.

✓ Correct, the will generate the remaining tacos after all guests get their equal share.

The following code segment has completed steps 1 & 2 of an algorithm that is designed to convert a person's height in inches only into feet and inches.

```
//Step 1: Declare variables
int height; //to store the height in inches only

//Step 2: Collect user input
cout << "How tall are you in inches? ";
cin >> height;
```

Use the following space to complete Step 3 and enter output statement(s) along with the appropriate expressions to display the person's height in feet and remaining inches. As a reminder, there are 12 inches in a foot.

```
cout << "You are " << height / 12 << " feet and " << height % 12 << " inches. \n";
```

Resubmit

Thank you for your response. Your output statement should look similar to this.



```
cout << "You are " << height / 12 << " feet and " << height % 12 << " inches. \n";
```

Describe another problem that could be solved using integer divisions.

```
Converting pounds into grams
```

Resubmit



Thanks for sharing.

Page 9 Assignment statements

Hotspot simple assignments

```
int inches_per_foot;
```

Given the above variable declaration, which of the following is a valid assignment statement?

- ☐ 12 = inches_per_foot;
- ☒ inches_per_foot = 12;
- ☐ Both of the above
- ☐ Neither of the above

✓ Correct; this would store 12 into the space reserved for inches_per_foot.

The following statements are not listed in a proper order. Rearrange them into proper order.

```
double miles = 4.5;
double km_per_mile = 1.60934;
double kilometers = miles * km_per_mile;
```

Check My Answer

✓ Correct, miles * km_per_mile must be listed AFTER the statements that assign values to miles and km_per_miles.

In your own words, explain what was wrong with the original order of the statements:

```
double kilometers = miles * km_per_mile;
double miles = 4.5;
double km_per_mile = 1.60934;
```

You cant find kilometers first if there's no stores variables with the proper values

Resubmit

✓ If an expression is set up before the variables in the expression have their values, the computer will not be able to accurately evaluate the expression. As a result, the variable `kilometers` will not have the right value.

Checkpoint named constants

In the following space, define a named constant to store the number of seconds per minute.

```
const int SECONDS_PER_MINUTE = 60
```

Resubmit

✓ const int SECONDS_PER_MINUTE = 60;

In the following space, define a named constant of your choice. Be sure to use descriptive name.

A constant to store the number of ounces per pounds

```
const int OUNCES_PER_POUND = 16
```

Resubmit

✓ Thanks for sharing.

LBD bank balance

In a banking software, the variable `bank_balance` is used to track how much money a customer has in their account.

```
double bank_balance;
```

Complete the assignment statement to set `bank_balance` initial value if a customer opens the account with \$400

```
bank_balance = 400;
```

✓ Correct. This allows the value 400 be assigned to the variable on the LHS of =.

```
double deposit;  
cout << "Enter the deposit amount:t";  
cin >> deposit;
```

Suppose the customer has come back to deposit \$80.26 to their account. The above code allows the program to collect the deposit amount.

Complete the following assignment statement to increase the `bank_balance` variable accordingly

```
bank_balance = bank_balance + deposit;
```

✓ Correct. This will allow the compiler to perform $400 + 80.26$ before updating the balance.

```
double withdrawal;  
cout << "Enter the withdrawal amount:t";  
cin >> withdrawal;
```

Suppose the customer has come to withdraw \$50 from the account instead of making a deposit. The above code allows the program to collect the deposit amount.

Complete the following assignment statement to increase the `bank_balance` variable accordingly

```
bank_balance = bank_balance - withdrawal;
```

✓ Correct! This would allow the compiler to perform $400 - 50$ before storing 350 to `bank_balance`.

LBD give changes

What value will be stored in `quarters` after line 6 is executed?

✓ Correct! $92 / 25$ returns 3

What value will be stored in `change` after line 7 is executed?

- ☒ 17
- ☐ 92
- ☐ 3
- ☐ 25

✓ Correct, $92 \% 25$ returns the remainder of the division

What value will be stored in `dimes` after line 9 is executed?

✓ Correct! $17 / 10$ returns the number of dimes needed.

Mark all of the following whose values are not affected by lines 6 - 12.

- ☒ DIME
- ☐ dimes
- ☐ nickels
- ☒ NICKEL

Check My Answer

✓ Correct, these are the named constants that would remain unchanged once their initial values are set.

Checkpoint updated algorithm pattern

What questions and/or tips for others do you have regarding the above program?

What's the best way to go about evaluating code?

Resubmit

✓ Thanks for sharing.

The above code segment has completed the first two steps needed to convert `weight` in ounces only into the corresponding `pounds` and `leftover_ounces`. For example, a weight of 50 ounces would be converted into 3 pounds and 2 leftover ounces while a weight of 40 ounces would be converted into 2 pounds and 8 leftover ounces.

In the following space, create an assignment statement to update the values of `pounds` based on `weight` and `OZ_PER_LB`.

```
cout << pounds = weight / OZ_PER_LB;
```

Resubmit

Thanks for sharing. Your assignment statement should look like this,

✓ `pounds = weight / oz_per_lb;`

In the following space, create an assignment statement to update the values of `leftover_ounces` based on `weight` and `OZ_PER_LB`.

```
cout << leftover_ounces = weight % OZ_PER_LB;
```

Resubmit

Thanks for sharing. Your assignment statement should look like this,

✓ `leftover_ounces = weight % oz_per_lb;`

Page 10 Type casting

LBD implicit type casting

```
0 | int quantity = 4.9;  
1 | double unit_cost = 10.00;  
2 | double total_cost = quantity * unit_cost;
```

What is the value stored in `quantity`?

✓ Correct! The conversion drops the decimal value.

What is the value stored in `total_cost`?

✓ Correct! The first statement assigns 4 to `quantity`.

Checkpoint Performance stat

Run the program by entering the same number for the goals attempted and goals made. What does the program show as the goal percentage?

✓ Correct. The program works well when the team has made goal for all their attempts.

Run the program by entering 5 as the goals attempted and 2 as the goals made. What does the program show as the goal percentage?

✓ Correct. The program does not works well when the team couldn't made goal for all their attempts. $2 / 5$ is an integer division. It returns 0 not 0.4.

Why do you think the program does not work well when the team couldn't make all of their goal attempts?

Because it's run as an integer not a double

Resubmit

✓ Thanks for sharing.

Describe another problem that you think could be solved using the similar approach of explicit type casting.

How many times it took to write a new program and successfully run it

Resubmit

✓ Thanks for sharing.