Turgut Alp Edis

21702587 Section 2

CS315 Homework #2

Short-Circuit Evaluation in Dart

1. How are the boolean values represented?

The boolean values are represented as false and true.

2. What operators are short-circuited?

The operators "||" (or) and "&&" (and) are short-circutied in Dart.

3. How are the results of short-circuited operators computed? (Consider also function calls)

When the operation is computed, for || operation if the left is true, the right is not computed and returns true, otherwise, the right is checked, and the value of right is returned. for && operation if the left is true, the result will be the right, otherwise the result will be the same as the left.

4. What are the advantages about short-circuit evaluation?

It avoids computationally complex tasks. It provides a check for the left without the runtime error for the right.

5. What are the potential problems about short-circuit evaluation?

It can cause unexpected results if it is not used properly. When the function uses memory, short-circuit may cause some unexpected results. Also, the time of the execution becomes longer since it adds new paths.

Short-Circuit Evaluation in JavaScript

1. How are the boolean values represented?

In JavaScript, Boolean is represented as true and false. Also, in integer scope, 1 and 0 can be used to represent true and false. Also, empty string and null is false.

2. What operators are short-circuited?

In JavaScript, OR "||" and AND "&&" operators are short-circuited.

3. How are the results of short-circuited operators computed? (Consider also function calls)

In Javascript, results are computed from left to right. For OR "||" operator, the result will be the first true value. If there is two values and left is true, then the result is left, otherwise right is the result. For AND "&&" operator, the result is the first false value. If there is two values and left is true, then the result is right, otherwise left is the result.

4. What are the advantages about short-circuit evaluation?

If the function gives some error or long calculations, short-circuit can prevent invoke of this function.

5. What are the potential problems about short-circuit evaluation?

Because of the short-circuit, whether called function is working or not could not be known. Also, the time of the execution becomes longer since it adds new paths.

Short-Circuit Evaluation in PHP

1. How are the boolean values represented?

In PHP, boolean is represented as true and false. However, when the value is printed, true printed as 1 and false printed as nothing.

2. What operators are short-circuited?

In PHP, OR "||”, AND "&&" operators are short-circuited.

3. How are the results of short-circuited operators computed? (Consider also function calls)

In PHP, results are computed from left to right. For "||" OR operator, the value is the left if the left is true, otherwise the value is the right. For "&&" AND operator, the value is the right if the left is true, otherwise the value is the left.

4. What are the advantages about short-circuit evaluation?

The long calculations and errors can be avoided.

5. What are the potential problems about short-circuit evaluation?

Because of the short-circuit, whether called function is working or not could not be known. Also, the time of the execution becomes longer since it adds new paths.

Short-Circuit Evaluation in Python

1. How are the boolean values represented?

In Python, boolean values are represented with "True" and "False". Also, all data types has boolean value with typecasting(coercion).

2. What operators are short-circuited?

The short-circuited operators are "and" and "or".

3. How are the results of short-circuited operators computed? (Consider also function calls)

The results computed left to right. For "or" operator, first, the left expression will be computed, if it is true, the result will be it. However, if it is false, the result will be the right expression. For "and" operator, first, the left expression is calculated, if it is true, then, right expression will be computed. However, when it is false, the result will be left expression.

4. What are the advantages about short-circuit evaluation?

It avoids computationally complex tasks. It provides a check for the left without the runtime error for the right.

5. What are the potential problems about short-circuit evaluation?

It can cause unexpected results if it is not used properly. When the function uses memory, short-circuit may cause some unexpected results. Also, the time of the execution becomes longer since it adds new paths.

Short-Circuit Evaluation in Rust

1. How are the boolean values represented?

In Rust, boolean values represented as True and False. Also, when true and false is type casted as integer, 1 displayed for true and 0 displayed for false.

2. What operators are short-circuited?

The && (and) and || (or) operators are short-circuited in Rust.

3. How are the results of short-circuited operators computed? (Consider also function calls)

For || operation, first, left is calculated and if it is true, the value will be left right won’t be calculated, otherwise the result will be the right. For && operation, first left is calculated and if it is true, the result will be the right, otherwise the result is left and right won’t be calculated.

4. What are the advantages about short-circuit evaluation?

It helps to avoid the unwanted situations such as errors and infinite loops.

5. What are the potential problems about short-circuit evaluation?

It can cause unexpected results if it is not used properly. When the function uses memory, short-circuit may cause some unexpected results. Also, the time of the execution becomes longer since it adds new paths.

Which language is the best for short-circuit evaluation considering advantages and disadvantages?

The logic of short-circuit is applied to the similar for all of the inspected languages (Dart, PHP, JavaScript, Python and Rust). Therefore, both the advantages and disadvantages are almost same. However, some of these languages, such as Python and Dart, the disadvantage of short-circuit is relevant to the memory since it may cause some unnecessary memory allocation, thus the memory leak and the program is crushed. On the other hand, for JavaScript and PHP, it causes the slow of the program. For the advantages, it is only the prevention of long executions and errors, and it is same for all languages. So, stating the different advantages is not possible. These five languages are same for the advantage. Thus, the evaluation can take place in terms of disadvantages. The two common disadvantages are the usage of additional memory and the extended execution time. When these are considered, the usage of additional memory may result more problems than the extended execution time since the memory leak may result in core dump while the extended execution time can result at most time waste in program testing. Then, the remaining languages for the evaluation are JavaScript and PHP. In terms of the usability, JavaScript is the best language for short-circuit evaluation since its boolean values can be written more easily than PHP.

How is this task completed? My Learning Strategy and Approach to This Homework

When I started the homework, initially I read the documentations, which aims to introduce the language, Boolean and short-circuit evaluation, for all languages since even if I did tutorship in Bilkent Python course whose code is CS115 and already know about the Boolean expressions, the short-circuit is almost completely new concept for me. So, I need to read Python documentation about the short-circuit. Also, though, I learnt the fundamentals of Dart language and wrote some example codes before, due to the short-circuit, I also need to read Dart evaluation documentations about short-circuit. Besides, since I had almost no knowledge about JavaScript, PHP, and Rust, I had to read the documentation of these three languages. For learning strategy, I read the documentation from the official websites such as MDN Web Docs, then I took a look at example codes from the lecture examples found on Dijkstra server and found on the educational websites such as W3Schools to learn how to code the specific language. After reading the documentations, I tried to implement the code to answer the short-circuit evaluation questions. Even if I received unexpected error or could not implement the code correctly, I searched the Stackoverflow for these issues and solved the error or wrong implementation problem. For instance, for JavaScript, I used MDN Web Docs (https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators, https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/A_first_splash) for booleans and short-circuit evaluation. For PHP, I used its official documentation website (https://www.php.net/manual/en/language.operators.logical.php) for Booleans and short-circuit evaluation. For Rust, I used official documentation website of Rust (https://doc.rust-lang.org/std/ops/index.html) for booleans and short-circuit evaluation. Since I had to learn just small part of the language, which are logical operators, boolean type and short-circuit evaluation logic, the learning time is relatively short. After gaining this knowledge, I started to write the code for explain the short-circuit evaluation questions. I used Visual Studio Code to write and run the code. Besides, I used terminal to run the codes because the extensions sometimes occurred with an error. Since I used Ubuntu as Operating System, working with terminal is easier than Windows since Ubuntu allows users to work more freely. While I was writing the code, I run the code regularly to test answers of the design problem questions. I tried to run possible answers for each question, and I tried to give an example about how the answer I gave is true. Though, I cannot demonstrate answer for question 5, which I took their answers from their documentations, since I could not show the current situation of the background system, I could show other answers for questions. Also, I tried to find multiple answers for each question as much as possible. After I finished the code, I copied it to the Dijkstra server and tested it in the server also to ensure that the code is working properly. When the code was compiled and run truly, I adjusted the comment sections to indicate the which line belongs to which question. Then, the code part of the homework finished.