

Workshop - 8

Workshop Value: 10 marks (4.375% of your final grade)

Please review the following documents:

1. Workshop [Grading Policies](#)
2. Workshop [Submission Procedures](#)
3. Workshop [Group Breakdown](#)

Workshop Overview

Black boxes are widely used and referred to in the world of programming. They represent specific processes that are hidden from the consumer (end-user, programmer, or conceptually by management etc.). In the programming sense, this can be a large-scale process (a system or application) or be as minute as a function (such as a shared Application Program Interface [API], or a web-service). There are two other types; **Grey Boxes** where some knowledge of how the process works is known and **White Boxes** where all the logic is visible/known by the consumer.

In the case of “**functions**”, each function should be limited to a very specific task. Although not necessary, most of the time functions will require some input information and use that input to produce some meaningful output (or result). This workshop is focused on function-based **Black Boxes** and **White Boxes** that do a very specific task and are called from other processes as required which will simplify the overall logic. The **Black Boxes** will be briefly described for you so you will know what functions exist and can implement into your solution and where you will not have to provide an algorithm for, while the **White Boxes** will be the sub-processes you must fully define.

Recap

Black Boxes are those that you do NOT have knowledge of how it works – you only know that it should accomplish some meaningful output based the input you provide it.

White Boxes are those that you have full knowledge of how it works because you will be defining those subprocesses and exposing all the logic (the algorithm) accordingly (ex: Logic-1, Logic-2, and Logic-3 components and any additional supportive subprocesses you may want to add).

Black and **White Boxes** can be called from anywhere in your defined processes – these are merely functions that will perform specific tasks to streamline/simplify the overall algorithm.

Workshop Details

You need to create a **software kiosk application** that will be used in a grocery store **self-checkout machine**. You need to define a solution that will provide a customer the means to process the items from their shopping cart (perishable and non-perishable products), through to the payment for the items that will include an itemized receipt (total breakdown of all the items).

Work Breakdown

Refer to the supplemental PDF document for this workshop: "**Workshop_08_Supplemental.pdf**"

Your Task

Individual Logic Assignment

1. Determine your individual assigned logic part based on your member# (see **Group Breakdown** link at the beginning of this document)
2. Where applicable, apply the core components of the **computational thinking** approach to problem solving to help you synthesize a solution
3. Submit your individual assigned part to your professor (see **Submission Procedures** link at the beginning of this document)

Group Solution

1. In the week the workshop is scheduled, you will be working in your assigned sub-group. See **Group Breakdown** link at the beginning of this document for details on how the sub-groups are determined.
2. Please review what is expected as described in the **Grading Policies** link at the beginning of this document.
3. Submit your group solution to your professor (if you are handing in physical paper answers, follow the directions as set by your professor, otherwise, refer to the **Submission Procedures** link at the beginning of this document)

Presentation

Decide among yourselves which member among you in the sub-group will be doing a presentation. Priority should be given to those who have not yet done one. Refer to the **Grading Policies**, and **Submission Procedures** links for details on deadlines, expectations and how to submit your work.