Money or the Box

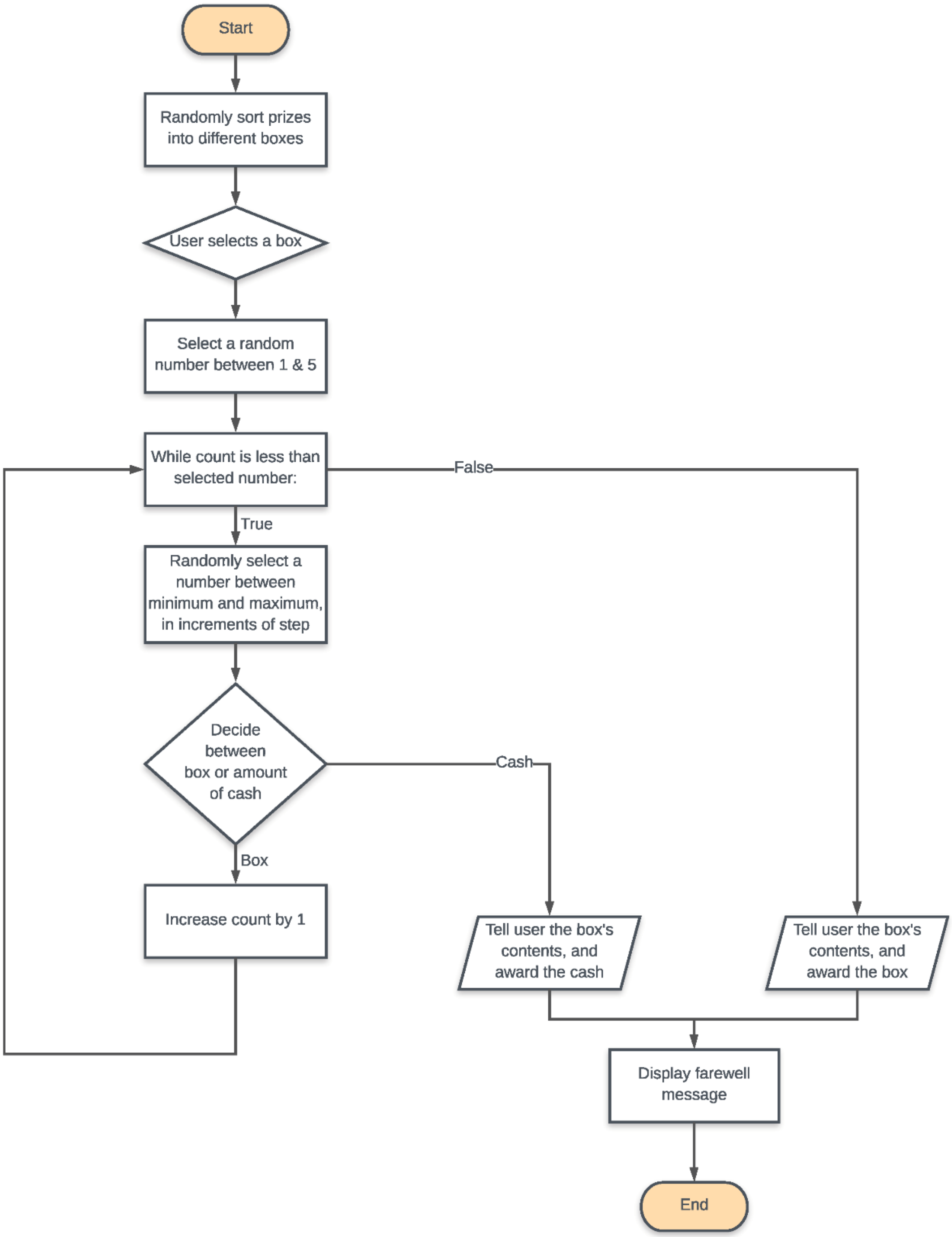
Purpose

The purpose of the brief is to create a python program that mimics the gameshow *Money or the Box.*

Target Language

This program will be created using Python 3.4.4.

Flow Chart


Start
Randomly sort prizes into different boxes
User selects a box
Select a random number between 1 & 5
While count is less than selected number:
False
True
Randomly select a number between minimum and maximum, in increments of step
Decide between box or amount of cash
Cash
Box
Increase count by 1
Tell user the box's contents, and award the cash
Tell user the box's contents, and award the box
Display farewell message
End

Updated Algorithm

```
START
    prizes = [“car”, “overseas holiday”, “TV”, “box of chocolates”, “DVD”]
    prizes = [“a car”, “an overseas holiday”, “a TV”, “a box of chocolates”, “a DVD”]
    SHUFFLE(prizes)
    boxes = []
    selectedBox = “”
    alreadyOffered = 0
    cashOrBox = “”
    DEFINE offerGenerator(offerMin, offerMax, offerStep) AS:
        RETURN RANDOM RANGE(offerMin, offerMax, offerStep)
        RETURN RANDOM RANGE(offerMin, offerMax + offerStep, offerStep)
    DEFINE userInput(rangeMaximum, inputMessage, ifError, exceptError) AS:
        inputValue = “”
        WHILE inputValue IS NOT INTEGER OR NOT IN RANGE(1, rangeMaximum + 1) DO:
            TRY THE FOLLOWING:
                inputValue = INTEGER INPUT(inputMessage)
                IF inputValue NOT IN RANGE(1, rangeMaximum + 1) DO:
                    PRINT(ifError)
            EXCEPT:
                PRINT(exceptError)
        RETURN inputValue
    PRINT(“Welcome to ‘Money or the Box’!”)
    FOR prize IN prizes DO:
        APPEND(boxes) WITH prize
    PRINT(“[NEW-LINE]Please select one of the following boxes: [NEW-LINE]”)
    FOR boxNumber IN RANGE(1, LENGTH(prizes) + 1) DO:
        PRINT(“[boxNumber]) Box #[boxNumber]”)
    selectedBox = userInput(LENGTH(prizes), “[NEW-LINE]Enter your selection: “, “[NEW-LINE]Your selection must be between 1 and [LENGTH(prizes)].”, “[NEW-LINE]Your selection must be an integer between 1 and [LENGTH(prizes)].”)
    numberOfOffers = RANDOM INTEGER(1, 5)
    WHILE alreadyOffered IS LESS THAN numberOfOffers OR cashOrBox IS NOT 1 DO:
    WHILE alreadyOffered IS LESS THAN numberOfOffers AND cashOrBox IS NOT 1 DO:
        IF alreadyOffered IS EQUAL TO 0 DO:
            currentOffer = offerGenerator(200, 500, 50)
        ELSE:
            currentOffer = currentOffer + offerGenerator(200, 5000, 100)
        PRINT("[NEW-LINE]Would you like to take the offer of $[currentOffer], or keep the box?[NEW-LINE][NEW-LINE]1) Give me the cash![NEW-LINE]2) I want the box!")
        cashOrBox = userInput(2, "[NEW-LINE]Enter your choice: ", "[NEW-LINE] Your choice must be either 1 or 2.", "[NEW-LINE]Your choice must be an integer that is either 1 or 2."
        alreadyOffered = alreadyOffered + 1
    IF cashOrBox IS EQUAL TO 1 DO:
        PRINT("[NEW-LINE]You have been given $[currentOffer], you have missed out on a [boxes[selectedBox - 1]].")
        PRINT("[NEW-LINE]You have been given $[currentOffer], you have missed out on a [prizes[selectedBox - 1]].")
    ELSE:
        PRINT("[NEW-LINE]Congratulations! You have won a [boxes[selectedBox - 1]]!")
        PRINT("[NEW-LINE]Congratulations! You have won a [prizes[selectedBox - 1]]!")
    PRINT("Thank you for playing 'Money or the Box'! Hope to see you again soon!")
END
```

Changes to Algorithm:

1) Changed values to add either 'a' or 'an', simply to ensure correct grammar when printing
2) Removed the boxes list, was unnecessary and inefficient as prizes could be directly used
3) Changed upper value of range to offerMax + offerStep as the upper boundary is exclusive
4) As with 2)
5) Changed the OR to AND, to ensure the code functioned as intended
6) Changed string to remove 'a', simply to ensure correct grammar when printing
7) As with 6)
8) Added [NEW-LINE] to beginning of string for correct and consistent formatting

Data Dictionary

Data item	Data type	Variable name	Initial value
List of potential prizes	List	prizes	“a car”, “an overseas holiday”, “a TV”, “a box of chocolates”, “a DVD”
The user’s selected box	Integer	selectedBox	
The number of cash offers already given to the user	Integer	alreadyOffered	0
The user’s choice of cash or box	Integer	cashOrBox	
Function to generate random offer amount	Function	offerGenerator	
Function to ensure user’s integer input is valid	Function	userInput	
The number of times an offer should be given to the user	Integer	numberOfOffers	Random integer between 1 and 5
The current offer amount	Integer	currentOffer	Randomly generated by offerGenerator

Test Plan

What is being tested	prizes	Test type	Expected output/result	Actual result
	“a car”, “an overseas holiday”, “a TV”, “a box of chocolates”, “a DVD”	Expected	List items have been shuffled around randomly	['a box of chocolates', 'a car', 'a DVD', 'a TV', 'an overseas holiday']
				['a TV', 'a box of chocolates', 'a car', 'an overseas holiday', 'a DVD']
				['a box of chocolates', 'a DVD', 'an overseas holiday', 'a car', 'a TV']
				['a DVD', 'an overseas holiday', 'a car', 'a TV', 'a box of chocolates']
				['an overseas holiday', 'a car', 'a box of chocolates', 'a TV', 'a DVD']

What is being tested	offerMin	offerMax	offerStep	Test type	Expected output/result	Actual result
	200	500	50	Expected	Random integer between 200 and 500 (inclusive), in increments of 50	300
						500
						350
						250
						450
	200	5000	100		Random integer between 200 and 5000 (inclusive), in increments of 100	400
						3200
						4700
						1000
						2300

What is being tested	Test type	Expected output/result	Actual result
🟪	Expected	Random integer between 1 and 5 (inclusive)	1
			4
			5
			5
			3

What is being tested	rangeMaximum	inputValue	Test type	Expected output/result	Actual result
🟩	LENGTH(prizes)	2	Expected	Passes through function, inputValue is returned successfully	Passes through function, inputValue is returned successfully
		1	Boundary		
		LENGTH(prizes)			
		0		Caught by IF statement with error message, WHILE loop triggers and input is re-done	'Your selection must be between 1 and 5.'
		LENGTH(prizes) + 1			
		LENGTH(prizes) + 4	Invalid		
		ABCD		Caught by EXCEPT with error message, WHILE loop triggers and input is re-done	'Your selection must be an integer between 1 and 5.'
🟩	2	1	Expected / Boundary	Passes through function, inputValue is returned successfully	Passes through function, inputValue is returned successfully
		2			
		0	Boundary	Caught by IF statement with error message, WHILE loop triggers and input is re-done	'Your choice must be either a 1 or 2.'
		3			
		6	Invalid		
		ABCD		Caught by EXCEPT with error message, WHILE loop triggers and input is re-done	'Your choice must be an integer that is either 1 or 2.'

What is being tested	alreadyOffered	numberOfOffers	cashOrBox	Test type	Expected output/result	Actual result
⬛	3	5	2	Expected	WHILE loop triggers	WHILE loop triggers
	3	5	1		Continue to IF statement	Continue to IF statement
	1	5	2	Expected / Boundary	WHILE loop triggers	WHILE loop triggers
	4	5	2			
	5	5	2	Boundary	Continue to IF statement	Continue to IF statement

What is being tested	cashOrBox	Test type	Expected output/result	Actual result
🟪	1	Expected	IF statement triggers	IF statement triggers
	2		Continue to ELSE statement	Continue to ELSE statement