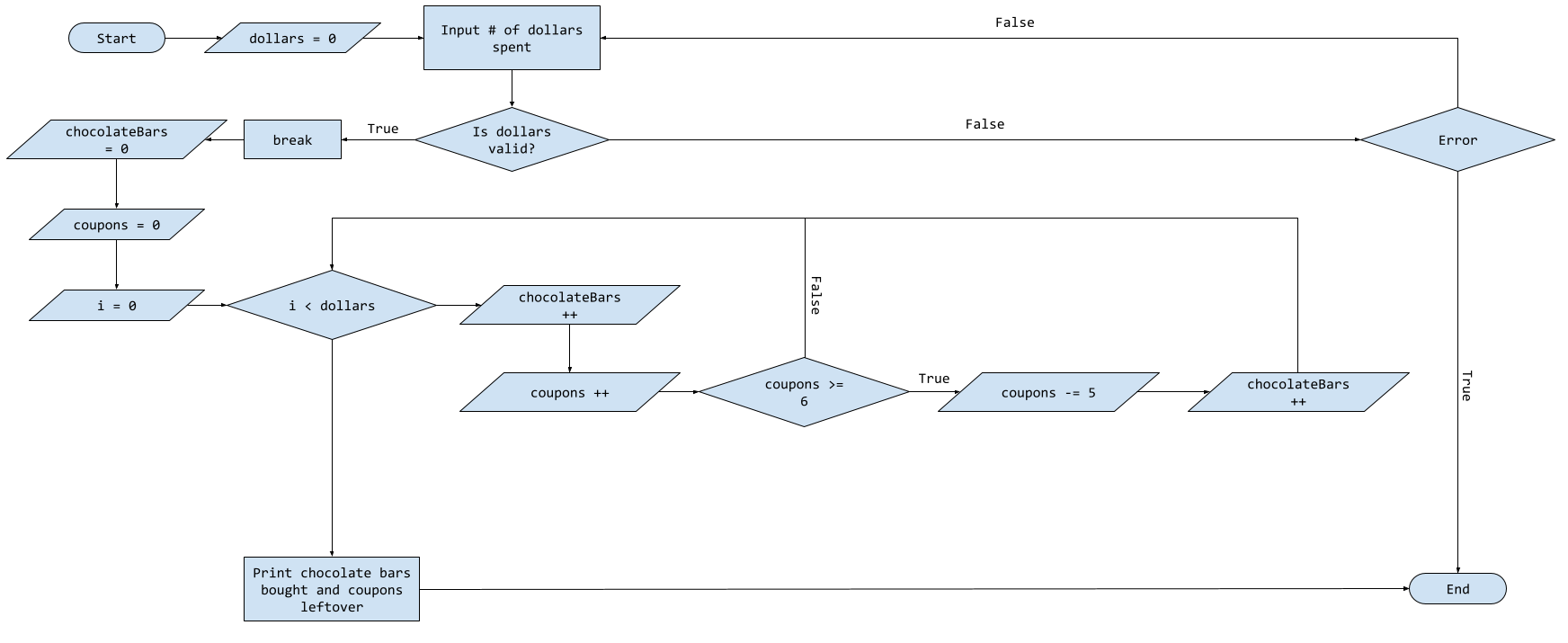
**Lab Report 05**

For this lab, we are given that a chocolate bar at a vending machine costs $1, and each purchase contains a coupon inside; Six coupons can then be redeemed for one chocolate bar. Using this information, we are asked to determine the amount of chocolate bars that can be purchased/redeemed with any number of dollars and number of coupons that are left over, if any.

I think that I can create a program to calculate the amounts by making use of for loops to iterate through code I will implement for the purchase for chocolate bars, for whatever amount of dollars is input by the user into the program. Within the loop, I can also check for the number of coupons held, which is incremented each time a chocolate bar is bought, to determine if an additional chocolate bar can be redeemed.



For this instance, the program will execute as normal because a valid number has been input by the user. 12 chocolate bars are bought – at $1 each – for $12, giving 12 coupons. Two additional bars can be redeemed for 6 coupons each, leaving 2 coupons since the redeemed bars also contained coupons.



Seeing as an invalid input of a negative integer was entered by the user in this instance, the code did not continue to allow for the user to enter a valid input.



In this final instance, however, a symbol was entered, which is invalid and caused the program to halt.



In my opinion, this lab was easier and less time consuming then I had imagined it would have been. Implementing the for loop for the purchasing of the chocolate bars was very simple and only took a little bit of time for me to figure out how to do.

Additional Questions:

1. When the condition is false: the body of a do-while loop runs 1 time.
2. What causes an infinite loop? How can infinite loops be avoided? In order for a loop to not execute infinitely, a condition for a break statement needs to be included so that it can end when a certain condition is met.
3. How is a while loop different from a do-while loop? A while loop checks its condition before it executes whereas a do-while loop checks it after the fact.
4. Given the following code snippet, what will the program print to the console?

int count = 1;

while (count < 10)

{

System.out.print(count + “\t”);

count ++;

}

1 2 3 4 5 6 7 8 9

1. Revise the following code snippet so that it uses a do-while loop instead of a while loop.

int count = 0;

while (count <= 15)

{

System.out.println(count);

count += 3;

}

int count = 0;

do

{

System.out.println(count);

count += 3;

}

while (count <= 15);