

Test Case #	Category	Type of Testing	Input	Output	Results
1	Expected	Black Box	4, 3	3	Pass
2	Unexpected	Black Box	A, 0	Invalid Input Try Again	Pass
3	Boundary/Expected	White Box	4, 16	0	Pass

```
1 *****
2 *   PROGRAMMED BY : Carlos Aguilera
3 *   CLASS         : CS1B
4 *   SECTION       : MW: 7:30p - 9:50p
5 *   LAB #7        : Testing
6 *****
7
8
9 Enter number of Sheep: 4
10 Enter number of Chocolate Bars: 3
11 Number of remaining Chocolate Bars: 3
12 Number of Chocolate Bars each sheep got: 0
13
14 Enter number of Sheep: A
15 Input was invalid try again!!
16 Enter number of Sheep: 0
17 Enter number of Chocolate Bars: 4
18 Number of remaining Chocolate Bars: 4
19 Number of Chocolate Bars each sheep got: 0
20
21 Enter number of Sheep: 4
22 Enter number of Chocolate Bars: 16
23 Number of remaining Chocolate Bars: 0
24 Number of Chocolate Bars each sheep got: 4
```

```
1 #pragma once
2 #include <iostream>
3 #include <iomanip>
4 #include <unistd.h>
5
6 void heading();//prints out heading
7 void handleInput(size_t &numOfSheeps, size_t &numOfChocolateBars);//handles
  input of sheeps and chocolate bars
8 void handleCalc(const size_t numOfSheeps, const size_t
  numOfChocolateBars);//handles the logic
9 void handleOutput(const size_t &remainder, size_t &i);//handles output
10 void handleOutput(const size_t &remainder);//overloaded function that handles
  output
```

```

1  /*****
2  * AUTHOR      : Carlos Aguilera
3  * STUDENT ID  : 1152562
4  * LAB #7     : Testing
5  * CLASS      : CS1B
6  * SECTION    : M-W
7  * DUE DATE   : 03.14.22
8  *****/
9
10 /*****
11 * Title: Testing
12 * -----
13 * FUNCTION:
14 *   This program calculates the number of chocolate bars a sheep should get
15 * -----
16 * Data Table
17 * -----
18 * size_t numOfSheeps IN & CALC - used to take input of number of sheeps and
19 * also a calc for how many chocolate bars a sheep should get
20 *
21 * size_t numOfChocolateBars IN & CALC - used to take input of number of
22 * chocolate bars and also a calc for how many chocolate bars a sheep should
23 * get
24 *****/
25 #include "../include/main.h"
26
27 int main() {
28     heading();
29
30     size_t numOfSheeps {};
31     size_t numOfChocolateBars {};
32
33     for (size_t i {0}; i < 3; i++) {
34         handleInput(numOfSheeps, numOfChocolateBars);
35         handleCalc(numOfSheeps, numOfChocolateBars);
36     }
37     return 0;
38 }

```

```

1 #include "../include/main.h"
2 /*****
3  * Title: handelInput
4  * -----
5  * FUNCTION:
6  *   this function handles input from the user and validates it if needed
7  *   I use a try catch for the validation
8  * -----
9  * Data Table
10 * -----
11 * bool inputValidated CALC - bool to say if input is validated or not
12 *****/
13
14
15 void handleInput(size_t &numOfSheeps, size_t &numOfChocolateBars) {
16     bool inputValidated = false;
17     std::cout << "\n";
18     do {
19         try {
20             std::cout << "Enter number of Sheep: ";
21             std::cin >> numOfSheeps;
22             if (std::cin.fail()) { //if cin failed meaning if they put something
invalid we clear cin and throw true and catch it
23                 std::cin.clear();
24                 std::cin.ignore(10, '\n');
25                 throw(true);
26             }
27             std::cout << "Enter number of Chocolate Bars: ";
28             std::cin >> numOfChocolateBars;
29             if (std::cin.fail()) {
30                 std::cin.clear();
31                 std::cin.ignore(10, '\n');
32                 throw(true);
33             }
34             inputValidated = true;
35         }
36         catch(bool invalid) {
37             std::cout << "Input was invalid try again!!\n";
38         }
39     }while(!inputValidated);
40 }

```

```

1 #include "../include/main.h"
2 /*****
3  * Title: handelCalc
4  * -----
5  * FUNCTION:
6  *   Handles the calculation of how many chocolate bars a sheep gets
7  * -----
8  * No Data Table
9  * -----
10 *****/
11
12
13 void handleCalc(const size_t numOfSheeps, const size_t numOfChocolateBars) {
14     /*
15      I used a for loop to test possible solutions on how many chocolate bars
16      a sheep got
17      if number of sheeps was greater than # of chocolate bars then the
18      remainder
19      was simply # of chocolate bars
20     */
21     for (size_t i {4}; i > 0; i--) {
22         if (numOfSheeps == 0 || numOfSheeps > numOfChocolateBars) {
23             handleOutput(numOfChocolateBars);
24             i = 1;
25         } else if (numOfChocolateBars >= (numOfSheeps * i)) {
26             size_t remainder {numOfChocolateBars - (numOfSheeps * i)};
27             handleOutput(remainder, i);
28         }
29     }
30 }

```

```

1 #include "../include/main.h"
2 /*****
3  * Title: handelOutput
4  * -----
5  * FUNCTION:
6  *   Handles output, also overloading this function
7  * -----
8  * No Data Table
9  * -----
10 *****/
11
12
13 void handleOutput(const size_t &remainder, size_t &i) {
14     std::cout << "Number of remaining Chocolate Bars: " << remainder << "\n";
15     std::cout << "Number of Chocolate Bars each sheep got: " << i << "\n";
16     i = 1;
17 }
18 void handleOutput(const size_t &remainder) {
19     std::cout << "Number of remaining Chocolate Bars: " << remainder << "\n";
20     std::cout << "Number of Chocolate Bars each sheep got: 0 \n";
21 }

```

```

1 #include "../include/main.h"
2
3 void heading() {
4     /*******
5     * CONSTANTS
6     * -----
7     * OUTPUT - USED FOR CLASS HEADING
8     * -----
9     * PROGRAMMER : Programmer's Name
10    * CLASS      : Student's Course
11    * SECTION    : Class Days and Times
12    * LAB_NUM    : Lab Number (specific to this lab)
13    * LAB_NAME   : Title of the Lab
14    *****/
15    const char PROGRAMMER[] = "Carlos Aguilera";
16    const char CLASS[]      = "CS1B";
17    const char SECTION[]    = "MW: 7:30p - 9:50p";
18    const int LAB_NUM       = 7;
19    const char LAB_NAME[]   = "Testing";
20
21    // (variable declarations go here)
22
23
24    /*******
25    * OUTPUT - Class Heading
26    *****/
27    std::cout << std::left;
28    std::cout << "*****\n";
29    std::cout << "*   PROGRAMMED BY : " << PROGRAMMER << std::endl;
30    std::cout << "*   " << std::setw(14) << "CLASS" << ": " << CLASS <<
std::endl;
31    std::cout << "*   " << std::setw(14) << "SECTION" << ": " << SECTION <<
std::endl;
32    std::cout << "*   LAB #" << std::setw(9) << LAB_NUM << ": " << LAB_NAME <<
std::endl;
33    std::cout << "*****\n\n";
34    std::cout << std::right;
35 }

```