Marilee Austin

CIT 260.4

L5 Individual Assignment

May 26, 2018

**double calcHuntingResource( hunger, itemQuantity, randomNumber)**

**Task**:

To determine if a hunt is successful and if so to select an animal based on number of bullets, hunger and random selection.

In a “resource scene” where the player can hunt, they first choose how many bullets they would like to use. That number is added to their hunger status and divided by a random number. Output number corresponds to the following results:

1. Squirrel 3lbs
2. Rabbit 10lbs
3. Deer 75lbs
4. Cougar 100lbs
5. Bear 220lbs
6. Bison 1000lbs
7. Unsuccessful 0lbs
8. Unsuccessful 0lbs
9. Unsuccessful 0lbs

Amount in pounds of animal is displayed

**Input:**

Number of bullets (itemQuantity)

Hunger (from HealthControl)

randomNumber

**Output:**

A number between 0 and 8. (Corresponds to a hunting outcome.)

**Validation Rules:**

If itemQuantity is less than 1 then the player will receive an error message

If item Quantity is greater than 100 then the player will receive an error message

If hunger is less than 0 the player will receive an error message

If hunger is greater than 5 the player will receive an error message

If randomNumber is less than 1 the player will receive an error message

If randomNumber is greater than 20 the player will receive an error message.

**Range:**

* 1-105 (100 is maximum #of bullets that can be purchased)
* Hunger is 0-5
* randomNumber is 1-20

**List:** Input cannot exceed the number of bullets in inventory.

**Format:** The items must be in any positive number format.

**Required:** Input is required.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **calcHuntingResource( hunger, itemQuantity, randomNumber)** | | | | | | | |  |
|  | Test Cases | | | | | | |
| Valid | Invalid | | | Boundary | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| **Inputs** | | | | | | | |
| (itemQuantity) | 10 | -1 | 45 | 72 | 1 | 100 | 50 |  |
| Hunger | 5 | 2 | 6 | 3 | 0 | 4 | 5 |  |
| randomNumber | 15 | 6 | 18 | -1 | 5 | 20 | 6 |  |
| **Outputs** | | | | | | | |  |
| result | 1 | -1 | -4 | -5 | 0 | 5 | 8 |
| **Error** |  | itemQuantity < 1 | hunger  > 5 | randomNumber  <1 |  |  |  |

**Note:**

Random Number Generator

import java.util.Random;

random rand = newRandom();

int n = rand.nestInt(20) + 1;

**Pseudo Code for**  **calcHuntingResource( hunger, itemQuantity, randomNumber)**

**BEGIN**

If (itemQuantity<1)

return -1

if (itemQuantity>100)

return -2

if (hunger < 0)

return -3

if (hunger > 5)

return -4

if (randomNumber < 1)

return -5

if (randomNumber > 20)

return -6

Double result = ((itemQuantity + hunger) / randomNumber)

return result.intValue();

**End**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Cases** | | | | | | | |
| **Variables** | **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| itemQuantity | 10 | -1 | 45 | 72 | 1 | 100 | 50 |
| hunger | 5 | 2 | 6 | 3 | 0 | 4 | 5 |
| randomNumber | 15 | 6 | 18 | -1 | 5 | 20 | 6 |
| **Output** | | | | | | | |
| result | 1 | -1 | -4 | -5 | 0 | 5 | 8 |