|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Function Name** | **#** | **Test Description** | **Sample Input** | **Expected Result** | **Actual Result** | **P/F** |
| addMore | 1 |  |  |  |  |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| addOns | 1 |  |  |  |  |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| breakdown | 1 | Testing an input that should output a 1 for all denominations | fChange = 1886.4 | Php 1000.00 - 1  Php 500.00 - 1  Php 200.00 - 1  Php 100.00 - 1  Php 50.00 - 1  Php 20.00 - 1  Php 10.00 - 1  Php 5.00 - 1  Php 1.00 - 1  Php 0.25 - 1  Php 0.10 - 1  Php 0.05 - 1 | Php 1000.00 - 1  Php 500.00 - 1  Php 200.00 - 1  Php 100.00 - 1  Php 50.00 - 1  Php 20.00 - 1  Php 10.00 - 1  Php 5.00 - 1  Php 1.00 - 1  Php 0.25 - 1  Php 0.10 - 1  Php 0.05 - 1 | **P** |
| 2 | Testing if the denomination will not display when the count is 0 | fChange = 520.05 | Php 500.00 - 1  Php 20.00 - 1  Php 0.05 - 1 | Php 500.00 - 1  Php 20.00 - 1  Php 0.05 - 1 | **P** |
| 3 | Testing if the input is a large number | fChange = 987654 | Php 1000.00 - 987  Php 500.00 - 1  Php 100.00 - 1  Php 50.00 - 1  Php 1.00 - 4 | Php 1000.00 - 987  Php 500.00 - 1  Php 100.00 - 1  Php 50.00 - 1  Php 1.00 - 4 | **P** |
| checkForPickups | 1 |  |  |  |  |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| checkLayout | 1 | Testing if the function will count the number of spaces correctly when ‘L’ is passed | cSelectedOpt = ‘L’  \*pLayout = {'L', 'P', 'L', '\0', 'O', '\0'} | 1 | 1 | **P** |
| 2 | Testing if the function will count the number of spaces correctly when ‘P’ is passed | cSelectedOpt = ‘P’  \*pLayout = {'L', 'P', 'L', '\0', 'O', '\0'} | 1 | 1 | **P** |
| 3 | Testing if the function will count the number of spaces correctly when ‘O’ is passed | cSelectedOpt = ‘P’  \*pLayout = {'L', 'P', 'L', '\0', 'O', '\0'} | 2 | 2 | **P** |
| 4 | Testing if the function will count the number of spaces correctly when other letters are passed | cSelectedOpt = ‘A’  \*pLayout = {'L', 'P', 'L', '\0', 'O', '\0'} | 2 | 2 | **P** |
| checkPosition | 1 | Testing if the function will display the error message for patch | \*pLayout = {'1', 'P', '2', '\0', 'O', '\0'}  nAns = 2; | Invalid. Already indicated to place a patch. | Invalid. Already indicated to place a patch. | **P** |
| 2 | Testing if the function will display the error message for logo | \*pLayout = {'1', 'P', '2', '\0', 'O', '\0'}  nAns = 4 | Invalid. Already indicated to place a logo. | Invalid. Already indicated to place a logo. | **P** |
| 3 | Testing if the function will display the error message for pocket | \*pLayout = {'1', 'P', '2', '\0', 'O', '\0'}  nAns = 2  cAns = ‘O’ | Invalid. Already indicated to place a pocket. | Invalid. Already indicated to place a pocket. | **P** |
| checkYorN | 1 | Testing an invalid input | cAns = ‘a’ | Invalid answer | Invalid answer | **P** |
| 2 | Testing a valid small letter input | cAns = ‘n’ | N | N | **P** |
| 3 | Testing a valid block letter input | cAns = ‘Y’ | Y | Y | **P** |
| chooseAddOn | 1 | Testing an invalid input | cAns = ‘a’ | Invalid choice | Invalid choice | **P** |
| 2 | Testing a valid small letter input | cAns = ‘p’ | P | P | **P** |
| 3 | Testing a valid block letter input | cAns = ‘L’ | L | L | **P** |
| chooseNeck | 1 | Testing an invalid input | cAns = ‘a’ | Invalid neckline | Invalid neckline | **P** |
| 2 | Testing a valid small letter input | cAns = ‘r’ | R | R | **P** |
| 3 | Testing a valid block letter input | cAns = ‘V’ | V | V | **P** |
| chooseQuantity | 1 | Testing a negative input | nAns = -5 | Invalid number of pieces | Invalid number of pieces | **P** |
| 2 | Testing an input less than 5 but not a negative number | nAns = 0 | Too few. Invalid number of pieces | Too few. Invalid number of pieces | **P** |
| 3 | Testing if the input is 5 | nAns = 5 | 5 | 5 | **P** |
| 4 | Testing a very large input | nAns = 987654 | 987654 | 987654 | **P** |
| chooseSize | 1 | Testing an invalid input | cAns = ‘a’ | Invalid size | Invalid size | **P** |
| 2 | Testing a valid small letter input | cAns = ‘s’ | S | S | **P** |
| 3 | Testing a valid block letter input | cAns = ‘X’ | X | X | **P** |
| computeAndDisplayPrice | 1 |  |  |  |  |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| computeAndDisplayTime | 1 |  |  |  |  |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| logoColor | 1 | Testing an invalid character input | cAns = ‘a’ | Invalid choice | Invalid choice | **P** |
| 2 | Testing an invalid numerical input | cAns = ‘0’ | Invalid choice | Invalid choice | **P** |
| 3 | Testing a valid input | cAns = ‘4’ | 4 | 4 | **P** |
| makeShirt | 1 | Testing if the correct dimension will be output based on the size | cSize = ‘S’ | nLength = 10  nWidth = 18 | nLength = 10  nWidth = 18 | **P** |
| 2 | Testing if the correct dimension will be output based on the size | cSize = ‘M’ | nLength = 12  nWidth = 22 | nLength = 12  nWidth = 22 | **P** |
| 3 | Testing if the correct dimension will be output based on the size | cSize = ‘X’ | nLength = 16  nWidth = 30 | nLength = 16  nWidth = 30 | **P** |
| makeShirtColumn | 1 |  | row = |  |  |  |
| 2 |  | row = |  |  |  |
| 3 |  | row = |  |  |  |
| order | 1 |  |  |  |  |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| paymentAndChange | 1 | Testing if the input is less than the total price | fTotal\_Price = 500  fCash = 499.99 | Amount entered is less than the total price. | Amount entered is less than the total price. | **P** |
| 2 | Testing if the input is equal to the total price | fTotal\_Price = 500  fCash = 500 | Total change: P 0.00  Thank you for paying the exact amount! | Total change: P 0.00  Thank you for paying the exact amount! | **P** |
| 3 | Testing if the input is negative | fCash = 0.5 | Amount entered is invalid. | Amount entered is invalid. | **P** |
| 4 | Testing if the input is greater than the total price | fTotal\_Price = 500  fCash = 500.05 | Total change: P 0.05  Breakdown of change: | Total change: P 0.05  Breakdown of change: | **P** |
| placement | 1 |  |  |  |  |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| printMainMenu | 1 | Testing an invalid character input | cAns = ‘a’ | Invalid choice | Invalid choice | **P** |
| 2 | Testing an invalid numerical input | cAns = ‘0’ | Invalid choice | Invalid choice | **P** |
| 3 | Testing a valid input | cAns = ‘4’ | 4 | 4 | **P** |
| printOrderMenu | 1 | Testing an invalid character input | cAns = ‘a’ | Invalid choice | Invalid choice | **P** |
| 2 | Testing an invalid numerical input | cAns = ‘6’ | Invalid choice | Invalid choice | **P** |
| 3 | Testing a valid input | cAns = ‘5’ | 5 | 5 | **P** |
| printPosition | 1 | Testing an input that should output “on left sleeve” | nPosition = 1 | on left sleeve | on left sleeve | **P** |
| 2 | Testing an input that should output “on bottom left” | nPosition = 5 | on bottom left | on bottom left | **P** |
| 3 | Testing an input that should output “on right chest” | nPosition = 2 | on right chest | on right chest | **P** |
| printReceipt | 1 |  |  |  |  |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| saveOrder | 1 |  |  |  |  |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |