# Homework: Test Levels and Test Types

## Unit Testing in the Real Life: Testing a Battery

|  |  |
| --- | --- |
| **Test #1** | Take a **bulb 1.5V** and check if the battery works as expected: the bulb should light up after connection properly. |
| **Test #2** | Take **multimeter** and check the **voltage**.   * It should be ~**1.5** volts. |
| **Test #3** | Take the battery and check it **visually**.   * Check its **length**. * Check its **diameter**. * Check if it is has a form of **cylinder**. * Check for **leakage, corrosion**, etc. |
| **Test #4** | Check with a compatible **flashlight**. This will check two things:   * Weather battery size matches the flashlight. * Weather the battery work as expected. (light be bulb) |
| **Test #5** | Check the **labels** on the battery.   * The denoted size should be “**AA**” * The denoted voltage should be “**1.5V**”. |
| **Test #6** | Check if “**+**” and “**-**” are correctly positioned using a multimeter. |
| **Test #7** | Environmental test:   * Low temperature, e.g. 2 degree Celsius. * High temperature, e.g. 45 degree Celsius. |
| **Test #8** | Check the expiration date label. It should be in the future. |
| **Test #8** | Overheating test. |

## Unit Testing in the Real Life: Testing a Light Bulb

|  |  |
| --- | --- |
| **Test #1** | Functionality Test: Verify that the bulb turns on and off when it should. |
| **Test #2** | Power Test: Test the bulb’s power consumption to ensure if falls within the specific range. |
| **Test #3** | Brightness Test: Verify that the bulb produces the specified amount of output. |
| **Test #4** | Color Temperature Test: Test the bulb’s temperature to ensure that it falls within the specified range. |
| **Test #5** | Heat Test: Test the bulb’s temperature to ensure that if does not get too hot or pose any safety hazards. |
| **Test #6** | Lifespan Test: Run the bulb for an extended period of time to ensure it last as long as it is supposed to. |
| **Test #7** | Compatibility Test: Test the bulb compatibility with different fixtures and lamp types. |
| **Test #8** | Durability Test: Test the bulb’s ability withstand physical shocks and vibrations. |
| **Test #9** | Flicker Test: Check the bulb for any flickering issue. |
| **Test #10** | Dimming Test: Test the bulb’s ability to dim smoothly and without any issue. |

## Unit Testing in the Software World: Age Checker

|  |  |  |
| --- | --- | --- |
| **#** | **Test Description** | **Pass / Fail** |
| **Test #1** | AgeChecker(0) 🡪 child | Pass |
| **Test #2** | AgeChecker(5) 🡪 child | Pass |
| **Test #3** | AgeChecker(12.9) 🡪 child | Pass |
| **Test #4** | AgeChecker(13) 🡪 teenager | Pass |
| **Test #5** | AgeChecker(19.5) 🡪 teenager | Pass |
| **Test #6** | AgeChecker(20) 🡪 adult | Pass |
| **Test #7** | AgeChecker(21) 🡪 adult | Pass |
| **Test #8** | AgeChecker(50) 🡪 adult | Pass |
| **Test #9** | AgeChecker(64.7) 🡪 adult | Pass |
| **Test #10** | AgeChecker(65) 🡪 elder | Pass |
| **Test #11** | AgeChecker(75.3) 🡪 elder | Pass |
| **Test #12** | AgeChecker(95) 🡪 elder | Pass |
| **Test #13** | AgeChecker(150) 🡪 elder. It shows “**error**” | Not Pass |
| **Test #14** | AgeChecker(15800) 🡪 error | Pass |
| **Test #15** | AgeChecker(-5) 🡪 error | Pass |
| **Test #16** | AgeChecker(-1) 🡪 error | Pass |
| **Test #17** | AgeChecker(“Peter”) 🡪 error | Pass |

## Unit Testing in the Software World: Income Checker

|  |  |  |
| --- | --- | --- |
| **#** | **Test Description** | **Pass / Fail** |
| **Test #1** | Income Checker(250) 🡪 “low” | Pass |
| **Test #2** | Income Checker(1500) 🡪 “mid” | Pass |
| **Test #3** | Income Checker(2999) 🡪 “mid” | Pass |
| **Test #4** | Income Checker(3000) 🡪 “high” | Pass |
| **Test #5** | Income Checker(5000) 🡪 “high” | Pass |
| **Test #6** | Income Checker(-500) 🡪 “error” | Pass |
| **Test #7** | Income Checker(999) 🡪 “low” | Pass |
| **Test #8** | Income Checker(1000) 🡪 “mid” | Pass |

## Integration Testing in the Real Life: Lighting the Bulb

|  |  |
| --- | --- |
| **Test #1** | Implement the **following circuit**, using the provided **components**:  A picture containing shape  Description automatically generated  The bulb should **light up**. |
| **Test #2** | Implement the **following circuit**, using the provided **components**:  Diagram  Description automatically generated  Switch on the switch button and The bulb should **light up**. |
| **Test #3** | Implement the **following circuit**, using the provided **components**:  Diagram  Description automatically generated  Switch off the switch button and The bulb should **not** **light up**. |

## \* Integration Testing in the Software World: Ads

|  |  |
| --- | --- |
| **Test #1** | Check after clicking on **[Log in]** button if it will redirect to appropriate Login form. |
| **Test #2** | Check after log in if it redirects to user home page. |
| **Test #3** | Check after log in with specific user name if the **[log out]** button says the same user name as yours. |
| **Test #4** | Check after log in if it redirects to anonymous home page. |
| **Test #5** | Check if it is possible to log in with invalid credentials. |
| **Test #6** |  |
| **Test #7** |  |
| **Test #8** |  |

## \* Integration Testing in the Software World: Credit Risk

Input ranges and respective credit risk:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **child** | **teenager** | **adult** | **elder** | **negative** |
| **low** | 100% | 80 | 55% | 60% | error |
| **mid** | 100% | 72 | 37 | 44 | error |
| **high** | 100% | 64 | 19 | 65 | error |
| **negative** | error | error | error | error | error |

Test cases with execution results:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **#** | **Test Description** | **Age risk** | **Income risk** | **Result** | **Pass / Fail** |
| **Test #1** | CreditRisk(age: 12, income: 10) 🡪 100% | Child 🡪 100% | low 🡪 50% | 100% | Pass |
| **Test #2** | CreditRisk(age: 12, income: 2500) 🡪100% | Child 🡪 100% | mid 🡪 30% | 100% | Pass |
| **Test #3** | CreditRisk(age: 12, income: 5000) 🡪100% | Child 🡪 100% | high 🡪 10% | 100% | Pass |
| **Test #4** | CreditRisk(age: -1, income: 100) 🡪error | Child 🡪 100% | low 🡪 50% | error | Pass |
| **Test #5** | CreditRisk(age: -1, income: 5000) 🡪error | Child 🡪 100% | high 🡪 10% | error | Pass |
| **Test #6** | CreditRisk(age: -1, income: -5000) 🡪error | Child 🡪 100% | high 🡪 10% | error | Pass |
| **Test #7** | CreditRisk(age: 18, income: 10) 🡪 80% | Teen 🡪 60% | low 🡪 50% | 80% | Pass |
| **Test #8** | CreditRisk(age: 18, income: 2500) 🡪 72% | Teen 🡪 60% | mid 🡪 30% | 72% | Pass |
| **Test #9** | CreditRisk(age: 18, income: 5000) 🡪 64% | Teen 🡪 60% | high 🡪 10% | 64% | Pass |
| **Test #10** | CreditRisk(age: 18, income: -5000) 🡪error | Teen 🡪 60% | low 🡪 50% | error | Pass |
| **Test #11** | CreditRisk(age: 35, income: 10) 🡪 55% | adult 🡪 10% | low 🡪 50% | 55% | Pass |
| **Test #12** | CreditRisk(age: 35, income: 2500) 🡪 37% | adult 🡪 10% | mid 🡪 30% | 37% | Pass |
| **Test #13** | CreditRisk(age: 35, income: 5000) 🡪 19% | adult 🡪 10% | High 🡪 10% | 19% | Pass |
| **Test #14** | CreditRisk(age: 70, income: 10) 🡪 60% | elder 🡪 20% | low 🡪 50% | 60% | Pass |
| **Test #15** | CreditRisk(age: 70, income: 2500) 🡪 44% | elder 🡪 20% | mid 🡪 30% | 44% | Pass |
| **Test #16** | CreditRisk(age: 70, income: 5000) 🡪 60% | elder 🡪 20% | high 🡪 10% | 60% | Pass |
| Regression test **#1** | CreditRisk(age: 12/18/35/70, income: 0) 🡪100%/ 80%/55%/60% | Child🡪 100%/Teen 🡪60% /adult🡪 10%/ elder 🡪 20% | low 🡪 50% | none | Not Pass |
| Regression test **#2** | CreditRisk(age: 0, income: 1000) 🡪100% | Child🡪 100% | low 🡪 50% | none | Not Pass |

## System Testing in the Real Life: Flashlight

|  |  |
| --- | --- |
| **Test #1** | Test switches on / switch off the light.   * Take the flashlight. Put new batteries correctly. Switch on the flashlight --> the bulb should light up. * Switch off the flashlight --> the bulb should light off. |
| **Test #2** | Test battery replacement. |
| **Test #3** | Test bulb replacement. |
| **Test #4** | Test battery duration. At least 1 hour of lighting with new batteries. |
| **Test #5** | Test the illumination distance. It should illuminate cleanly at distance of 30 meters or less (with new batteries). |
| **Test #6** | Shock resistance test: Fall from the table and check if it still works correctly. |
| **Test #7** | Operation under high / low temperature. |
| **Test #8** | Overheat test. |
| **Test #9** | Water resistance test. |

## System Testing in the Real Life: Digital Scale

|  |  |
| --- | --- |
| **Test #1** | Accuracy Test: Test the scale’s accuracy by weighing standard weights to ensure measurements are within the specific range. |
| **Test #2** | Calibration Test: Verify that the scale can be calibrated properly to ensure accurate readings. |
| **Test #3** | Overload Test: Test the scale’s ability to handle weights above its maximum capacity to ensure it does not break or malfunctions. |
| **Test #4** | Repeatability Test: Test the scale’s ability to give consistent readings for the same weight multiple times. |
| **Test #5** | Linearity Test: Verify that the scale measures weights accurately across its entire range, and that the readings are proportional to the actual weights. |
| **Test #6** | Battery Test: Test the scale’s battery to ensure it last as long as specified. |
| **Test #7** | Environmental Test: Test the scale’s ability to withstand different environmental conditions, such as temperature and humidity, without affecting its accuracy or functionality. |

## System Testing in the Software World: Number Calculator

|  |  |  |
| --- | --- | --- |
| **#** | **Test Description** | **Pass / Fail** |
| **Test #1** | Calculate(5, +, 10) 🡪 15 | Pass |
| **Test #2** | Calculate(2000000000000, +, 5) 🡪 2000000000005 | Fail |
| **Test #3** | Calculate(10x, +, man) 🡪 invalid input | Pass |
| **Test #4** | Calculate(10, -, 5) 🡪 5 | Pass |
| **Test #5** | Calculate(10, \*, 5) 🡪 50 | Pass |
| **Test #6** | Calculate(1.5, \*,1 5) 🡪 2.25 | Pass |
| **Test #7** | Calculate(2e2, \*,2e2) 🡪 4000 | Pass |
| **Test #8** | Calculate(100000000e10000000000\*,1000000e10000000000) 🡪 infinity | Pass |
| **Test #9** | Calculate(infinity /, infinity) 🡪 invalid calculation | Pass |
| **Test #10** | Calculate(10, /, 5) 🡪 2 | Pass |

## Acceptance Testing in the Real Life: Flashlight

|  |  |
| --- | --- |
| **Test #1** | The customer takes the flashlight, **switch on / off** the light, and assures it works. |
| **Test #2** | The customer checks the flash **illumination**. |
| **Test #3** | The customer checks how easy it is to **replace the batteries.** |

## Acceptance Testing in the Real Life: Digital Scale

|  |  |
| --- | --- |
| **Test #1** | Accuracy Test: Test the scale’s accuracy by weighing a range of standard weights and comparing the results to their actual weights. |
| **Test #2** | Performance Test: Test the scale’s performance under various conditions, such as temperature and humidity, without affecting its accuracy or functionality. |
| **Test #3** | Durability Test: Test the scale’s durability by subjecting it to different levels of stress such as dropping it from different heights or exposing it to different chemicals. |
| **Test #4** | Compliance Testing: Test the scale’s compliance with relevant regulations and standards. |

## Acceptance Testing in the Software World: Number Calculator

|  |  |
| --- | --- |
| **Test #1** | Test the calculator by executing with valid integers. |
| **Test #2** | Test the calculator by executing with valid decimals numbers. |
| **Test #3** | Test the calculator by executing with exponential numbers. |
| **Test #4** | Test the calculator by executing with infinity. |
| **Test #5** | Test the calculator by executing with very large numbers. |
| **Test #6** | Test the calculator by executing with very small numbers. |

## Functional and Non-Functional Tests: Flashlight

|  |  |
| --- | --- |
| **Functional Tests** | **Non-Functional Tests** |
| Switching on and off | **Test buttery duration** |
| Test Battery | Illumination distance |
| Test bulb replacement | overheating |
|  | Water resistance test |