

**FUNCTIONS**

Software Testing

**EMPLOYEE’S PAYROLL**

**(b)**

FUNCTIONS FROM THE CASE STUDY:

·        GetEmployeeData():

This function will ask the user to enter the data of employee; their id, name, hours worked, pay rate and the number of holidays.

·        WeeklyWage(int hour\_worked, int pay\_rate, int holidays):

This function will calculate the weekly wage of the employee depending upon the data entered.

·        TaxDeduction(double wage):

This function will calculate the tax deduction depending upon their weekly wage.

·        InsuranceDeduction(double wage):

This function is dependent on the tax payed by the employee.

**(c)**

Using worst case BVA, identify test cases of the program

For worst BVA 5n =53 =125 test cases.

**FOR FUNCTION:**

* WeeklyWage(int hour\_worked, int pay\_rate, int holidays):

For hour 40<= x >=167

For payrate 20<= x >=70

For holidays 1<=x>=7

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Case | Hour Worked | Pay rate | Holiday | Expected Output |
| 1 | 20 | 20 | 2 | Error |
| 2 | 20 | 20 | 3 | Error |
| 3 | 20 | 20 | 4 | Error |
| 4 | 20 | 20 | 6 | Error |
| 5 | 20 | 20 | 7 | Error |
| 6 | 20 | 30 | 2 | 600 |
| 7 | 20 | 30 | 3 | 300 |
| 8 | 20 | 30 | 4 | 206 |
| 9 | 20 | 30 | 6 | 0 |
| 10 | 20 | 30 | 7 | -100 |
| 11 | 30 | 40 | 2 | 1200 |
| 12 | 30 | 40 | 3 | 900 |
| 13 | 30 | 40 | 4 | 800 |
| 14 | 30 | 40 | 6 | 600 |
| 15 | 30 | 40 | 7 | 500 |
| 16 | 30 | 49 | 2 | 1470 |
| 17 | 30 | 49 | 3 | 1170 |
| 18 | 30 | 49 | 4 | 1070 |
| 19 | 30 | 49 | 6 | 870 |
| 20 | 30 | 49 | 7 | 770 |
| 21 | 40 | 50 | 2 | 2000 |
| 22 | 40 | 50 | 3 | 1700 |
| 23 | 40 | 50 | 4 | 1600 |
| 24 | 40 | 50 | 6 | 1400 |
| 25 | 40 | 50 | 7 | 1300 |
| 26 | 40 | 60 | 2 | Error |
| 27 | 40 | 60 | 3 | 0 |
| 28 | 40 | 60 | 4 | 2100 |
| 29 | 40 | 60 | 6 | 2000 |
| 30 | 40 | 60 | 7 | 1800 |
| 31 | 50 | 31 | 2 | 2015 |
| 32 | 50 | 31 | 3 | 1715 |
| 33 | 50 | 31 | 4 | 1615 |
| 34 | 50 | 31 | 6 | 1415 |
| 35 | 50 | 31 | 7 | 1315 |
| 36 | 50 | 70 | 2 | Error |
| 37 | 50 | 70 | 3 | Error |
| 38 | 50 | 70 | 4 | Error |
| 39 | 50 | 70 | 6 | Error |
| 40 | 50 | 70 | 7 | Error |
| 41 | 60 | 49 | 2 | 4410 |
| 42 | 60 | 49 | 3 | 4110 |
| 43 | 60 | 49 | 4 | 4010 |
| 44 | 60 | 49 | 6 | 3810 |
| 45 | 60 | 49 | 7 | 3710 |
| 46 | 60 | 50 | 2 | 4500 |
| 47 | 60 | 50 | 3 | 4200 |
| 48 | 60 | 50 | 4 | 4100 |
| 49 | 60 | 50 | 6 | 3900 |
| 50 | 60 | 50 | 7 | 3800 |
| 51 | 70 | 30 | 2 | 3450 |
| 52 | 70 | 30 | 3 | 3150 |
| 53 | 70 | 30 | 4 | 3050 |
| 54 | 70 | 30 | 6 | 2850 |
| 55 | 70 | 30 | 7 | 2750 |
| 56 | 70 | 31 | 2 | 3565 |
| 57 | 70 | 31 | 3 | 3265 |
| 58 | 70 | 31 | 4 | 3165 |
| 59 | 70 | 31 | 6 | 2965 |
| 60 | 70 | 31 | 7 | 2665 |
| 61 | 80 | 40 | 2 | 5600 |
| 62 | 80 | 40 | 3 | 5300 |
| 63 | 80 | 40 | 4 | 5200 |
| 64 | 80 | 40 | 6 | 5000 |
| 65 | 80 | 40 | 7 | 4900 |
| 66 | 80 | 49 | 2 | 6860 |
| 67 | 80 | 49 | 3 | 6560 |
| 68 | 80 | 49 | 4 | 6460 |
| 69 | 80 | 49 | 6 | 6260 |
| 70 | 80 | 49 | 7 | 6160 |
| 71 | 90 | 50 | 2 | 8250 |
| 72 | 90 | 50 | 3 | 7950 |
| 73 | 90 | 50 | 4 | 7850 |
| 74 | 90 | 50 | 6 | 7650 |
| 75 | 90 | 50 | 7 | 7550 |
| 76 | 90 | 30 | 2 | 4950 |
| 77 | 90 | 30 | 3 | 4650 |
| 78 | 90 | 30 | 4 | 4550 |
| 79 | 90 | 30 | 6 | 4350 |
| 80 | 90 | 30 | 7 | 4250 |
| 81 | 100 | 31 | 2 | 5890 |
| 82 | 100 | 31 | 3 | 5590 |
| 83 | 100 | 31 | 4 | 5490 |
| 84 | 100 | 31 | 6 | 5290 |
| 85 | 100 | 31 | 7 | 5190 |
| 86 | 100 | 40 | 2 | 7600 |
| 87 | 100 | 40 | 3 | 7300 |
| 88 | 100 | 40 | 4 | 7200 |
| 89 | 100 | 40 | 6 | 7000 |
| 90 | 100 | 40 | 7 | 6900 |
| 91 | 110 | 49 | 2 | 10535 |
| 92 | 110 | 49 | 3 | 10235 |
| 93 | 110 | 49 | 4 | 9935 |
| 94 | 110 | 49 | 6 | 9835 |
| 95 | 110 | 49 | 7 | 9735 |
| 96 | 110 | 50 | 2 | 10750 |
| 97 | 110 | 50 | 3 | 10450 |
| 98 | 110 | 50 | 4 | 10350 |
| 100 | 110 | 50 | 6 | 10150 |
| 101 | 120 | 50 | 7 | 10050 |
| 102 | 120 | 30 | 2 | 12000 |
| 103 | 120 | 30 | 3 | 6900 |
| 104 | 120 | 30 | 4 | 6800 |
| 105 | 120 | 30 | 6 | 6600 |
| 106 | 120 | 30 | 7 | 6500 |
| 107 | 120 | 31 | 2 | 7440 |
| 108 | 120 | 31 | 3 | 7140 |
| 109 | 120 | 31 | 4 | 7040 |
| 110 | 120 | 31 | 6 | 6840 |
| 111 | 130 | 31 | 7 | 7515 |
| 112 | 130 | 40 | 2 | 10600 |
| 113 | 130 | 40 | 3 | 10300 |
| 114 | 130 | 40 | 4 | 10200 |
| 115 | 130 | 40 | 6 | 10000 |
| 116 | 130 | 40 | 7 | 9900 |
| 117 | 130 | 49 | 2 | 12985 |
| 118 | 140 | 49 | 3 | 13910 |
| 119 | 140 | 49 | 4 | 13810 |
| 120 | 140 | 49 | 6 | 13610 |
| 121 | 150 | 49 | 7 | 14735 |
| 122 | 150 | 50 | 2 | 15750 |
| 123 | 150 | 50 | 3 | 15450 |
| 124 | 160 | 50 | 4 | 16600 |
| 125 | 160 | 50 | 6 | 16400 |

* TaxDeduction(double wage)

Wage x>=10000

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Wage** | **Marital Status** | **Expected Output** |
| 1 | 15000 | Married | 7000 |
| 2 | 18000 | Married | 6000 |
| 3 | 20000 | Married | 4000 |
| 4 | 70000 | Married | 5600 |
| 5 | 56000 | Married | 1230 |
| 6 | 15000 | Single | 5422 |
| 7 | 18000 | Single | 1111 |
| 8 | 20000 | Single | 9999 |
| 9 | 70000 | Single | 4533 |
| 10 | 56000 | Single | 6522 |

* Insurance(Tax)

Tax x>=1000

|  |  |  |
| --- | --- | --- |
| Case | Tax | Expected Output |
| 1 | 70000 | 69000 |
| 2 | 40000 | 39000 |
| 3 | 23000 | 22000 |
| 4 | 45555 | 44555 |
| 5 | 12121 | 11121 |

File submitted titled “EmployeeTest.Java”



Strong Robust Equivalence Class

We will check the values beyond the boundaries of the ranges

For hour 40<= x >=167

For payrate 20<= x >=70

For holidays 1<=x>=7

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Case | Hour Worked | Pay rate | Holiday | Expected Output |
| 1 | 20 | 20 | 2 | Error |
| 2 | 20 | 20 | 3 | Error |
| 3 | 20 | 20 | 4 | Error |
| 4 | 20 | 20 | 6 | Error |
| 5 | 20 | 20 | 7 | Error |
| 6 | 20 | 30 | 2 | 600 |
| 7 | 20 | 30 | 3 | 300 |
| 8 | 20 | 30 | 4 | 206 |
| 9 | 20 | 30 | 6 | 0 |
| 10 | 20 | 30 | 7 | -100 |
| 11 | 30 | 40 | 2 | 1200 |
| 12 | 30 | 40 | 3 | 900 |
| 13 | 30 | 40 | 4 | 800 |
| 14 | 30 | 40 | 6 | 600 |
| 15 | 30 | 40 | 7 | 500 |
| 16 | 30 | 49 | 2 | 1470 |
| 17 | 30 | 49 | 3 | 1170 |
| 18 | 30 | 49 | 4 | 1070 |
| 19 | 30 | 49 | 6 | 870 |
| 20 | 30 | 49 | 7 | 770 |
| 21 | 40 | 50 | 2 | 2000 |
| 22 | 40 | 50 | 3 | 1700 |
| 23 | 40 | 50 | 4 | 1600 |
| 24 | 40 | 50 | 6 | 1400 |
| 25 | 40 | 50 | 7 | 1300 |
| 26 | 40 | 60 | 2 | Error |
| 27 | 40 | 60 | 3 | 0 |
| 28 | 40 | 60 | 4 | 2100 |
| 29 | 40 | 60 | 6 | 2000 |
| 30 | 40 | 60 | 7 | 1800 |
| 31 | 50 | 31 | 2 | 2015 |
| 32 | 50 | 31 | 3 | 1715 |
| 33 | 50 | 31 | 4 | 1615 |
| 34 | 50 | 31 | 6 | 1415 |
| 35 | 50 | 31 | 7 | 1315 |
| 36 | 50 | 70 | 2 | Error |
| 37 | 50 | 70 | 3 | Error |
| 38 | 50 | 70 | 4 | Error |
| 39 | 50 | 70 | 6 | Error |
| 40 | 50 | 70 | 7 | Error |
| 41 | 60 | 49 | 2 | 4410 |
| 42 | 60 | 49 | 3 | 4110 |
| 43 | 60 | 49 | 4 | 4010 |
| 44 | 60 | 49 | 6 | 3810 |
| 45 | 60 | 49 | 7 | 3710 |
| 46 | 60 | 50 | 2 | 4500 |
| 47 | 60 | 50 | 3 | 4200 |
| 48 | 60 | 50 | 4 | 4100 |
| 49 | 60 | 50 | 6 | 3900 |
| 50 | 60 | 50 | 7 | 3800 |
| 51 | 70 | 30 | 2 | 3450 |
| 52 | 70 | 30 | 3 | 3150 |
| 53 | 70 | 30 | 4 | 3050 |
| 54 | 70 | 30 | 6 | 2850 |
| 55 | 70 | 30 | 7 | 2750 |
| 56 | 70 | 31 | 2 | 3565 |
| 57 | 70 | 31 | 3 | 3265 |
| 58 | 70 | 31 | 4 | 3165 |
| 59 | 70 | 31 | 6 | 2965 |
| 60 | 70 | 31 | 7 | 2665 |
| 61 | 80 | 40 | 2 | 5600 |
| 62 | 80 | 40 | 3 | 5300 |
| 63 | 80 | 40 | 4 | 5200 |
| 64 | 80 | 40 | 6 | 5000 |
| 65 | 80 | 40 | 7 | 4900 |
| 66 | 80 | 49 | 2 | 6860 |
| 67 | 80 | 49 | 3 | 6560 |
| 68 | 80 | 49 | 4 | 6460 |
| 69 | 80 | 49 | 6 | 6260 |
| 70 | 80 | 49 | 7 | 6160 |
| 71 | 90 | 50 | 2 | 8250 |
| 72 | 90 | 50 | 3 | 7950 |
| 73 | 90 | 50 | 4 | 7850 |
| 74 | 90 | 50 | 6 | 7650 |
| 75 | 90 | 50 | 7 | 7550 |
| 76 | 90 | 30 | 2 | 4950 |
| 77 | 90 | 30 | 3 | 4650 |
| 78 | 90 | 30 | 4 | 4550 |
| 79 | 90 | 30 | 6 | 4350 |
| 80 | 90 | 30 | 7 | 4250 |
| 81 | 100 | 31 | 2 | 5890 |
| 82 | 100 | 31 | 3 | 5590 |
| 83 | 100 | 31 | 4 | 5490 |
| 84 | 100 | 31 | 6 | 5290 |
| 85 | 100 | 31 | 7 | 5190 |
| 86 | 100 | 40 | 2 | 7600 |
| 87 | 100 | 40 | 3 | 7300 |
| 88 | 100 | 40 | 4 | 7200 |
| 89 | 100 | 40 | 6 | 7000 |
| 90 | 100 | 40 | 7 | 6900 |
| 91 | 110 | 49 | 2 | 10535 |
| 92 | 110 | 49 | 3 | 10235 |
| 93 | 110 | 49 | 4 | 9935 |
| 94 | 110 | 49 | 6 | 9835 |
| 95 | 110 | 49 | 7 | 9735 |
| 96 | 110 | 50 | 2 | 10750 |
| 97 | 110 | 50 | 3 | 10450 |
| 98 | 110 | 50 | 4 | 10350 |
| 100 | 110 | 50 | 6 | 10150 |
| 101 | 120 | 50 | 7 | 10050 |
| 102 | 120 | 30 | 2 | 12000 |
| 103 | 120 | 30 | 3 | 6900 |
| 104 | 120 | 30 | 4 | 6800 |
| 105 | 120 | 30 | 6 | 6600 |
| 106 | 120 | 30 | 7 | 6500 |
| 107 | 120 | 31 | 2 | 7440 |
| 108 | 120 | 31 | 3 | 7140 |
| 109 | 120 | 31 | 4 | 7040 |
| 110 | 120 | 31 | 6 | 6840 |
| 111 | 130 | 31 | 7 | 7515 |
| 112 | 130 | 40 | 2 | 10600 |
| 113 | 130 | 40 | 3 | 10300 |
| 114 | 130 | 40 | 4 | 10200 |
| 115 | 130 | 40 | 6 | 10000 |
| 116 | 130 | 40 | 7 | 9900 |
| 117 | 130 | 49 | 2 | 12985 |
| 118 | 140 | 49 | 3 | 13910 |
| 119 | 140 | 49 | 4 | 13810 |
| 120 | 140 | 49 | 6 | 13610 |
| 121 | 150 | 49 | 7 | 14735 |
| 122 | 150 | 50 | 2 | 15750 |
| 123 | 150 | 50 | 3 | 15450 |
| 124 | 160 | 50 | 4 | 16600 |
| 125 | 160 | 50 | 6 | 16400 |
| 126 | 0 | 30 | 1 | Error |
| 127 | 0 | 30 | 3 | Error |
| 128 | 0 | 30 | 4 | Error |
| 129 | 0 | 30 | 5 | Error |
| 130 | 0 | 30 | 7 | Error |
| 131 | 170 | 50 | 1 | Error |
| 132 | 170 | 50 | 3 | Error |
| 133 | 170 | 50 | 4 | Error |
| 134 | 170 | 50 | 5 | Error |
| 135 | 170 | 50 | 7 | Error |
| 136 | 60 | 10 | 1 | Error |
| 137 | 60 | 10 | 3 | Error |
| 138 | 60 | 10 | 4 | Error |
| 139 | 60 | 10 | 5 | Error |
| 140 | 60 | 10 | 7 | Error |
| 141 | 30 | 80 | 1 | Error |
| 142 | 30 | 80 | 3 | Error |
| 143 | 30 | 80 | 4 | Error |
| 144 | 30 | 80 | 5 | Error |
| 145 | 30 | 80 | 7 | Error |
| 146 | 111 | 24 | 0 | Error |
| 147 | 89 | 44 | 0 | Error |
| 148 | 32 | 48 | 0 | Error |
| 149 | 67 | 67 | 0 | Error |
| 150 | 32 | 45 | 0 | Error |
| 151 | 88 | 56 | 9 | Error |
| 152 | 66 | 33 | 10 | Error |

* TaxDeduction(double wage)

Wage x>=10000

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Wage** | **Marital Status** | **Expected Output** |
| 1 | 15000 | Married | 7000 |
| 2 | 18000 | Married | 6000 |
| 3 | 20000 | Married | 4000 |
| 4 | 70000 | Married | 5600 |
| 5 | 56000 | Married | 1230 |
| 6 | 15000 | Single | 5422 |
| 7 | 18000 | Single | 1111 |
| 8 | 20000 | Single | 9999 |
| 9 | 70000 | Single | 4533 |
| 10 | 56000 | Single | 6522 |
| 11 | 9000 | Married | Error |
| 12 | 5000 | Married | Error |
| 13 | 3444 | Married | Error |
| 14 | 1000 | Married | Error |
| 15 | 5674 | Married | Error |
| 16 | 9000 | Single | Error |
| 17 | 5000 | Single | Error |
| 18 | 3444 | Single | Error |
| 19 | 1000 | Single | Error |
| 20 | 5647 | Single | Error |

* Insurance(Tax)

Tax x>=1000

|  |  |  |
| --- | --- | --- |
| Case | Tax | Expected Output |
| 1 | 70000 | 69000 |
| 2 | 40000 | 39000 |
| 3 | 23000 | 22000 |
| 4 | 45555 | 44555 |
| 5 | 12121 | 11121 |
| 6 | 4000 | Error |
| 7 | 3000 | Error |
| 8 | 2000 | Error |

From strong robust equivalence class, we can conclude that it makes you check beyond the boundaries the limit defined and as a result we only get errors. However, making robust equivalence class is an extra effort as in weak BVA there were 125 test cases and in strong robust there are 152 and the extra test cases are only letting us know that the error will be generated.