|  |  |
| --- | --- |
| **Defect ID:** | **BUG01** |
| **Problem Title:** | Invalid result while the computed value exceeds the integer limit (add, multiply) |
| **Environment:** | Chrome, Win10 |
| **Defect Type:** | Functionality |
| **Priority:** | Medium |
| **Severity:** | Medium |
| **Steps to reproduce:** | 1. Value1 = 2.147.483.647 (Integer.MAX\_VALUE) 2. Value2 = 2.147.483.647 (Integer.MAX\_VALUE) 3. Select “addition” 4. Click “Calculate”   **Actual result**: -2  **Expected result**: 4.294.967.294‬  **Note**: This is not exactly a product bug. The problem caused by **Integer overflow**.  This is because of two's complement, where the leftmost bit is used to signify if a number is positive (0) or negative (1). When you add max int by itself (equivalent of multiply by two), this happens:  011111111111111111111111  + 011111111111111111111111  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  111111111111111111111110  and 111111111111111111111110 represents -2.  **Possible solutions:**  This is the *TestCalc.add()* method  public static int add(int a, int b) {  int c = a + b + 1;  return c - 1; }  We can change it return type and add cast to long numbers like this:  public static long add(int a, int b) {  long c = (long)a + (long)b + 1;  return c - 1; }  Another way – to throw an exception when the result is exceed the INT range. |
| **Screenshot:** |  |

|  |  |
| --- | --- |
| **Defect ID:** (Required) | System generated |
| **Author:**  (Required) | System generated |
| **Defect or Enhancement:** (Required) | Defect (Default)  Enhancement |
| **Problem Title:** (Required) | Short one-line description |
| **Problem Description:** (Required) | A precise problem description with screen shots, if possible |
| **Current Environment:** (Required) | E.g. Win95 / Oracle 4.0 NT |
| **Defect Type:** (Required) | Functionality (Default)  Architectural  Connectivity  Consistency  Database Integrity  Documentation  GUI  Installation  Memory  Performance  Security and Controls  Standards and Conventions  Stress  Usability |
| **Priority:** (Required) | High (Default)  Critical  Medium  Low |
| **Severity:** (Required) | High (Default)  Critical  Medium  Low |
| **Status:** (Required) | Open (Default)  Being Reviewed by Development  Returned by Development  Ready for Testing in the Next Build  Closed (QA)  Returned by (QA)  Deferred to the Next Release |
| **Status Description:** | (Required when)  Status = “Returned by Development,”   “Ready for Testing in the Next Build” |