|  |  |  |  |
| --- | --- | --- | --- |
| Name | Matrikelnummer | E-Mailadresse | Studiengang |
| Tobias Hoge | 439 224 | [t\_hoge03@uni-muenster.de](mailto:t_hoge03@uni-muenster.de) | Wirtschaftsinformatik |
| Fabian Ingenhorst | 441 378 | [f\_inge03@uni-muenster.de](mailto:f_inge03@uni-muenster.de) | Wirtschaftsinformatik |
| Ann-Cathrin Mutz | 438 871 | a\_mutz01@uni-muenster.de | Wirtschaftsinformatik |

Aufgabe 7)

1. 7/2 = 3

Begründung: Im verwendeten Datentyp (Integer) können nur ganze Zahlen behandelt werden. Hier wird dann automatisch abgerundet 3,5 -> 3

1. 4/2.0 = 2.0

Begründung: Bei einer Rechnung von Integer-Werten mit Double-Werten wird automatisch in den „allgemeineren“ Datentyp konvertiert. In diesem Fall ist das Ergebnis dann vom Typ Double und hat dadurch die Form 2.0

1. 0/0.0 = NaN (Not a Number)

Begründung: Diese Rechenoperation ist mathematisch nicht definiert. In dem IEEE-754 Standard ist definiert, dass in diesem Fall kein Error auftritt sondern der Wert „NaN“ ausgegeben werden soll.

1. 42/0 =

Begründung: Diese Rechenoperation ist mathematisch nicht definiert. Da der Datentyp Integer aber nicht den Wert „NaN“ annehmen kann kommt es hier zu einer Fehlermeldung und einem Abbruch des Programms.

1. 42/0.0 = Infinity

Begründung: Bei dem Wert 0.0 handelt es sich um eine Zahl, die der 0 sehr nahe kommt aber aufgrund von Rundungsfehlern nicht tatsächlich den Wert 0 hat. Hier dividiert man also die Zahl 42 durch eine sehr kleine Zahl, was dazu führt, dass das Ergebnis gegen Unendlich läuft.

1. i\*i = -1486618624

Begründung: Der Datentyp Integer besteht aus 32 Bit, wobei das erste Bit angibt, ob die Zahl positiv oder negativ ist. Bei dieser Berechnung werden mehr als 32 Bit benötigt und das führt dazu, dass sich das erste Bit verändert und das Ergebnis damit negativ wird.

1. i\*l = 1000000000000000000

Begründung: Bei der Multiplikation von Integer mit Long wird das Ergebnis in den Datentyp Long konvertiert. Hier kann eine solche Zahl ohne Probleme dargestellt werden.

1. l\*l\*l = -6930898827444486144

Begründung: Hier tritt das gleiche Problem auf wie bei Aufgabe f). Der Datentyp Long hat nicht genug Bits um das Ergebnis korrekt darzustellen.