Bài 1: Khối lệnh thực hiện:

Khai báo kiểu biến của t là Integer, đồng thời gán cho t giá trị của biến a. Sau đó gán cho biến b giá trị của biến t, và gán cho biến a giá trị của biến b.

Chẳng hạn, nếu ta nhập vào a = 1 và b = 2, thì nó sẽ trả về giá trị là: a = 1, b = 1, c = 1.

Bài 2:

!(a < b) và !(a > b) trả về kết quả ngược lại với điều kiện trong biểu thức.

!(a < b) thì trả về a>= b.

!(a > b) thì trả về a<=b.

* (!(a<b) && !(a>b)) có ý nghĩa là a = b.

Bài 3:

Vì khi lấy hai số nguyên chia cho nhau, ta được 1 số nguyên. Muốn ra kết quả 3.3333, thì ta phải khai báo biến của a = 10, b = 10 là kiểu số thực, và kết quả cũng phải ở dạng số thực. Hoặc để 10 và 3 vẫn ở dạng Int, nhưng kết quả là dạng số thực thì nhân thêm 1.0 ta sẽ được 3.333333

Bài 4: Sửa thành:

Double force = (G \* mass1 \* mass2)/( r \* r);

Bài 5: Sửa thành

boolean isA = ( 90 <= grade && grade <= 100);

Khi ta viết như đề bài, Java sẽ tính toán 90 <= x trước rồi trả về giá trị là “true” hoặc “false”, rồi lấy cái true false đó đi so sánh với cái đằng sau, nên khi biên dịch sẽ xảy ra lỗi.

Bài 6:

Kết quả sẽ ra False.

Bài 7:

Khi đó, Complier sẽ hiện lỗi bad operand types for binary operator ‘\*’ vì 27 là kiểu Int, nhưng three lại là kiểu String.

Bài 8:

Complier xuất hiện lỗi: “ Biến x có lẽ đã chưa được khởi tạo”, và lỗi ở dòng println.