Worksheet 02 - Relations/Conditionals & Branching

a) Evaluate each of the following expression below, and indicate its value in the box provided. Be sure to specify the appropriate type of the result (e.g., `7.0` rather than `7` for a `double`, Strings in `"quotes"`, `true` or `false` for a `boolean`).

```
(2 <= 2) && (5 >= 6)
```

```
= true && false
= false
```

```
(!true) || (!false)
```

```
= false || true
= true
```

```
((4>=3)||(5+6==12))&&(5/2==2)
```

```
= ( true || (11 == 12) ) && (2==2)
= ( true || false ) && (true)
= true && true
= true
```

b) Complete the following fragment of code so that the boolean "isDivBy10" will be true if num1 is evenly divisible by 10 (false otherwise), while the boolean "isDivBy4" will only be true if num1 is not divisible by 10 but is divisible by 4. Do not use any IF statements.

```
long  num1;

// assume num1 is assigned to here (not shown)


boolean isDivBy10;          // true if integer part of num1
                            // is evenly divisible by 10

isDivBy10 = (num1 % 10 == 0);



boolean isDivBy4;           // true if num1 is not divisible
                            // by 10 but is divisible by 4


isDivBy4 = !isDivBy10 && (num1 % 4 == 0);
```

c) What value is assigned to fee by the if statement below, when speed is 75? Does this seem appropriate? Would you modify it (if so how)?

```
double fee;

if (speed > 35) {
    fee = 20.0;
}
else if (speed > 50) {
    fee = 40.00;
}
else if (speed > 75) {
    fee = 60.00;
}
```

```
double fee = 0;

if (speed > 75) {
    fee = 60.0;
}
else if (speed > 50) {
    fee = 40.00;
}
else if (speed > 35) {
    fee = 20.00;
}
```

fee = 20.0 -> probably a logic error (should be fee=40.0)
-> also could add no fee if not speeding >35? (see fixes above)

d) Given two temperatures a and b, a method `icyHot(a,b)` returns `true` if one is less than 0 and the other is greater than 100. Complete the code for the method `icyHot` (below)

e.g.

```
    icyHot(120, -1) → true
    icyHot(-1, 120) → true
    icyHot(2, 120) → false
```

```
boolean icyHot(int temp1, int temp2) {

      boolean t1NegT2Over = temp1<0 && temp2>100;
      boolean t2NegT1Over = temp2<0 && temp1>100;

      return (t1NegT2Over || t2NegT1Over);



      // another solution (you decide which is more readable?)
      // return (temp1 * temp2 < 0) && abs(temp1 - temp2) >= 102;

      // → interpret this solution as:
      // "one temp is negative (i.e. -1 or less), and their diff is at least 102
      // i.e. the one that is greater must be 101 or higher)"

}
```

e) Assume that a, b, and c are variables of type int. Consider the following three conditions:

i. `(a == b) && (a == c) && (b == c)`
ii. `(a == b) || (a == c) || (b == c)`
iii. `(a - b) * (b - c) * (a - c) == 0`

Which of the conditions above is always true if the values of at least two of the variables a, b, and c are equal? (circle the best answer)

A. i. only

B. ii. only

C. iii. Only

D. i. and ii. Only

E. ii. and iii. only *(only need one relation to be true in (ii) for it to be true; if any two vars are same, there will be at least one zero in the expression on LHS of (iii), which will make it equal to 0)*