# OUTPUT TRACE TABLE

**Program Source File:
 Directions are listed on the next page – read them all before starting!!**

| Line # | OUTPUT PRODUCED | Value of Variables (include the name!) |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## OUTPUT:

(Ignore this table until it's covered in lecture – please leave it blank until then)

| Line # | Program Output(please list this exactly as it appears on the screen)(NOT what is listed in the program source code) |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |

## Directions:

 **You should remove these directions before handing in this file for homework assignments, etc. Failure to do so will result in a loss of points.**

 The objective of Output Tracing (as opposed to Program (execution) Tracing) is to develop your ability to quickly, informally trace a program’s execution without having to keep track of every single detail in meticulous detail.

When you’re output tracing, you should maintain a sheet of scratch paper that you use to keep track of the values of any variables that the program creates. In the above table you should list any output that the program produces, in the exact same order that the program produces it. Whenever a line of output is produced, you should record the line number of the program that produced the output, and also include a quick comment about what the program was doing when the output was produced. Note that since you're simulating what the program would produce on-screen, you should NOT include the double-quotes ( " ) that we use in the Java source code to indicate where the text starts or ends

 Here’s an example of tracing program output:

| Line | Program Source Code |
| --- | --- |
| 1 | import becker.robots.\*; |
| 2 |  |
| 3 | public class ICE\_01\_Tutorial\_1 extends Object |
| 4 | { |
| 5 |  public static void main(String[] args) |
| 6 |  {  |
| 12 |  System.out.println("PROGRAM STARTS HERE!"); |
| 13 |  int x = 0; |
| 14 |  while( x < 2 ) |
| 15 |  { |
| 16 |  System.out.println("x is: " + x ); |
| 17 |  x = x + 1; |
| 18 |  } |
| 19 |  System.out.println("PROGRAM ENDS HERE!"); |
| 20 |  } |
| 21 | } |

| Line # | OUTPUT PRODUCED | What’s Happening Near The Line That Causes The Output (Quick summary, in your own words) |
| --- | --- | --- |
| 12 | **PROGRAM STARTS HERE!** | Pre-loop message is printed |
| 16 | **x is 0** | First time through the loop |
| 16 | **x is 1** | Second time through the loop |
| 19 | **PROGRAM ENDS HERE!** | After-loop message is printed |