CS 1324 Spring 2021 Homework 12 Nested Loops

Jordan McFadden

TOTAL POINTS

20 / 20

QUESTION 1

1 Question 1 5 / 5

✓ **- 0 pts Correct**

- 1 pts first is not correct (aside from missing endpoint)

- 0 pts Last value of first missing or extra value

- 2 pts values in second only repeat once or more than five times instead of three times

- 1 pts values in second only repeat twice or repeat four times, instead of three times

- 0 pts Last value of second missing or extra value

- 2 pts Product contains three or more errors

- 1 pts Product contains one or two errors

- 0 pts Arithmetic mistake

- 5 pts Blank or incorrect

- 3 pts Not an infinite loop

QUESTION 2

2 Question 2 5 / 5

✓ **- 0 pts Correct**

- 1 pts Two or more errors in count

- 0 pts count missing or extra value

- 1 pts Two or more errors in other

- 0 pts other missing or extra last value(s)

- 1 pts Two or more errors in data[other]

- 0 pts data[other] has a missing or extra value

- 1 pts Two or more errors in index

- 0 pts index missing or extra last value

- 1 pts Two or more errors in data[index]

- 0 pts data[index] has a missing or extra value

- 5 pts Blank or incorrect

- 3 pts Not an infinite loop

QUESTION 3

3 Question 3 5 / 5

✓ **- 0 pts Correct**

- 2 pts Two or more mistakes in sum

- 1 pts One mistake in sum

- 1 pts Two or more mistakes in first

- 0 pts Last value of first not included

- 2 pts Three or mores mistakes in second

- 1 pts One or two mistakes in second

- 0 pts Last value(s) in each iteration missing in second

- 3 pts Not an infinite loop

- 10 pts Blank or incorrect

QUESTION 4

4 Question 4 5 / 5

✓ **- 0 pts Correct**

- 5 pts Blank or incorrect

- 1 pts next missing three or more values (aside from missing endpoints)

- 0 pts next missing last value in each iteration

- 2 pts Two or more wrong values in data

- 1 pts One wrong value in data

- 2 pts minIndex two or more values missing or incorrect

- 1 pts minIndex has one value missing or incorrect

- 0 pts temp incorrect or missing (other than missing last value)

- 5 pts Blank or incorrect

gradescope

Homework 12: Nested Loops

CS 1323/4 Spring 2021

Name: Jordan McFadden

Student ID (usually 112-XXX-XXXX or 113-XXX-XXXX): 113502650

1. (5 points) Trace the nested loop below in the table. If either of the loops is an infinite loop, trace three iterations and write "infinite loop" in the table.

```
int product=1;
for (int first = 1; first < 4; ++first)
{
      for (int second = 1; second < 3; ++second)
      {
            product = product * second;
      }
}
```

first	second	product
1	1	1
2	2	1
3	3	2
4	1	2
	2	4
	3	4
	1	8
	2	
	3	

2. (5 points) Trace the nested loop below in the table. If either of the loops is an infinite loop, trace three iterations and write "infinite loop" in the table.

```
int[] data ={5, 7, 2, 4, 1, 6};

int count = 0;
for (int index = 0; index < data.length; ++index)
{
      for (int other = index+1; other < data.length; ++other)
      {
            if (data[other] - data[index] > 1)
                  ++count;
      }
}
```

count	other	data[other]	index	data[index]
0	1	7	0	5
1	2	2	1	5
2	3	4	2	5
3	4	1	3	5
4	5	6	4	5
5	6	2	5	7
	2	4	6	7
	3	1		7
	4	6		7
	5	4		2
	6	1		2
	3	6		2
	4	1		4
	5	6		4
	6	6		1
	4			
	5			
	6			
	5			
	6			
	6			

3. (5 points) Trace the nested loop below in the table. If either of the loops is an infinite loop, trace three iterations and write “infinite loop” in the table.

```
int sum = 0;
for(int first = 4; first > -3; --first)
{
      int second = 1;
      while (second < first)
      {
            sum = sum + first;
            ++second;
      }
}
```

sum	first	second
0	4	1
4	3	2
8	2	3
12	1	4
15	0	1
18	-1	2
20	-2	3
	-3	1
		2
		1
		1
		1
		1

4. (5 points) Trace the nested loop below in the table. If either of the loops is an infinite loop, trace three iterations and write “infinite loop” in the table.

```
int[] data = {7, 1, 5, 2, 6, 4};
for (int index=0; index < data.length-1; ++index)
{
     int minIndex = index;
     for (int next = index+1; next < data.length; ++next)
     {
          if (data[minIndex] > data[next])
          // data[next] is smaller than data[minIndex]
          {
               minIndex = next;
          }
     }

     int temp = data[index];
     data[index] = data[minIndex];
     data[minIndex] = temp;

}
```

data[0]	data[1]	data[2]	data[3]	data[4]	data[5]	index	minIndex	next	temp
7	1	5	2	6	4	0	0	1	7
1	7	5	2	6	4	1	1	2	7
1	2	5	7	6	4	2	1	3	5
1	2	4	7	6	5	3	2	4	7
1	2	4	5	6	7	4	3	5	6
1	2	4	5	6	7	5	2	6	
							5	2	
							3	3	
							4	4	
							5	5	
							4	6	
								3	
								4	
								5	
								6	
								4	
								5	
								6	
								5	
								6	