

Big-O Worksheet 1 Name Tanmai Kalisipudi

Instructions: The following algorithms will display a different number of stars, depending on the value in nNum. Classify the Big-O growth rate, and justify your answer with a calculation.

<u>notation</u>	<u>name</u>	<u>effect of doubling data</u>
O(1)	constant	no effect
O(log n)	logarithmic	increases work by 1
O(n)	linear	doubles work
O(n · log n)	sometimes called "linearithmic" or "supralinear"	2x < work < 4x
O(n ²)	quadratic	quadruples work
O(2 ⁿ)	exponential	if data increases by 1, work doubles
O(n!)	factorial	if data increases by 1, work > 2x

1. How many stars would be displayed by the following algorithm? 100
How many stars would be displayed if nNum were doubled? 400

```
int nNum = 10;
for(int nOuter = 0; nOuter < nNum; nOuter++)
    for(int nInner = 0; nInner < nNum; nInner++)
        System.out.print("*");
```

Big-O notation: O(n²) Justify your answer: Since its doubling the work, its quadratic.

2. How many stars would be displayed by the following algorithm? 10
How many stars would be displayed if nNum were doubled? 20

```
int nNum = 10;
for(int nOuter = 0; nOuter < nNum; nOuter++)
    System.out.print("*");
```

Big-O notation: O(n) Justify your answer: Since you double the work when the number is doubled it is linear.

3. How many stars would be displayed by the following algorithm? 4
How many stars would be displayed if nNum were doubled? 5

```
int nNum = 10;
for(int nOuter = 1; nOuter <= nNum; nOuter*=2)
    System.out.print("*");
```

Big-O notation: O(log n) Justify your answer: It increases by 1.

4. How many stars would be displayed by the following algorithm? 1
How many stars would be displayed if nNum was doubled? 1

```
int nNum = 10;
char[] caStars = new char[100];
for(int nI = 0; nI < 100; nI++)
    caStars[nI] = '*';
System.out.print("" + caStars[nNum]);
```

Big-O notation: O(1) Justify your answer: It doesn't depend on

nNum and its going

100 times every time

5. How many stars would be displayed by the following algorithm? 32
 How many stars would be displayed if nNum were increased by 1? 64

```
int nNum = 5;
int nStars = 1;
while (nNum > 0)
{
    nStars *= 2;
    nNum--;
}
for (int nI = 0; nI < nStars; nI++)
    System.out.print("*");
```

Big-O notation: $O(2^n)$ Justify your answer: exponentially growing

6. How many stars would be displayed by the following algorithm? 40
 How many stars would be displayed if nNum were doubled? 100

```
int nNum = 10;
for (int nOuter = 1; nOuter <= nNum; nOuter++)
    for (int nInner = 1; nInner <= nNum; nInner *= 2)
        System.out.print("*");
```

Big-O notation: $O(n \cdot \log n)$ Justify your answer: two nested for
(one linear, one log)

7. How many stars would be displayed by the following algorithm? 10
 How many stars would be displayed if nNum were doubled? 20

```
public static void main(String[] args) {
    int nNum = 10;
    printStar(nNum);
}
public static void printStar(int nNum) {
    if (nNum > 0) {
        System.out.print('*');
        printStar(nNum - 1);
    }
}
```

Big-O notation: $O(n)$ Justify your answer: simple counting

8. How many stars would be displayed by the following algorithm? 4
 How many stars would be displayed if nNum were doubled? 5

```
public static void main(String[] args) {
    int nNum = 10;
    printStar(nNum);
}
public static void printStar(int nNum) {
    if (nNum > 0) {
        System.out.print('*');
        printStar(nNum / 2);
    }
}
```

Big-O notation: $O(\log n)$ Justify your answer: cut in half each
time