Java
Recursion Worksheet #1

Name –
Period –

1. Trace the following program and show the separate stack frames (i.e. the values of the parameters and/or local variables in each stack frame). Show the final output in the rectangle provided as well.

```
public class RecursionWorksheet1Ex1
{
   public static void main(String[] args)
   {
      int num = 10;
      System.out.println(mystery(num));
   }

   public static int mystery(int myNum)
   {
      if (myNum < 2)
         return 3;
      else
         return 2 + mystery(myNum - 2);
   }
}
```

Output: 13

m(10)
$\boxed{11}$
2 + m(8)
$\boxed{9}$
2 + m(6)
$\boxed{7}$
2 + m(4)
$\boxed{5}$
2 + m(2)
2 + m(0)
3

2. What is the output of the program in Exercise #1 if the original value of the variable num is 1 instead of 10? 3

3. Trace the following program and show the separate stack frames (i.e. the values of the parameters and/or local variables in each stack frame). Show the final output in the rectangle provided as well.

```
public class Ch17Worksheet1Ex3
{
   public static void main(String[] args)
   {
      int num = 13;
      System.out.println(mystery(num));
   }

   public static int mystery(int myNum)
   {
      if (myNum <= 0)
         return 0;
      else if (myNum % 2 == 0)
         return 2 + mystery(myNum - 1);
      else
         return 1 + mystery(myNum - 2);
   }
}
```

Output: 7

m(13)
↓
1 + m(11)
↓
1 + m(9)
↓
1 + m(7)
↓
1 + m(5)
↓
1 + m(3)
↓
1 + m(1)
↓
1 + m(-1)
0

Java
Recursion Worksheet #3

Name –
Period –

1. Compute `mystery(20)` where `mystery` is the following recursive method. Show scratchwork for full credit.

```
public static int mystery(int num)
{
     if (num > 0)
          return (num + mystery(num - 1));

     return 0;
}
```

m(20)
↓
20 + m(19)
↓
19 + m(18)
↓
....

$$\sum_{i=1}^{20} i$$

$$\frac{20}{2}(1 + 20)$$

$$10(21) = 210$$

2. Rewrite `mystery` as an iterative function as simply and as efficiently as possible.

```
public static int mystery(int num)
{
    int sum =0;
    for (int i = num; i > 0; i--) {
         sum + = num;
    }
    return sum;
}
```

space complexity much lower.

b(64) ↓   p64^

b(32) ↓   p32^

b(16) ↓   p16^

b(8) ↓   p8^

b(4) ↓   p4^

b(2) ↓   p2^

b(1)   return

Name: ___________________
AP Computer Science
Recursion Worksheet

1) What is the output?

```
public class RecursionWorksheet {
     public static void main(String[] args) {
          blah(64);
     }//end main
     public static void blah(int n){
          if(n%2!=0)
               return;
          else{
               System.out.print(n+" ");
               blah(n/2);
          }
     }//end blah
}
```

0
2⟌1
0
1

prints : 64^32^16^8^4^2^

2) Rewrite the function above to reverse the order of the numbers that are output.

★ move Sys ... after blah(n/2)

f(hello) o

+ f(hell) l

↓+ f(hel) l

+

3) What is the output of the following code? f(he) e

l h

```
public class RecursivePractice {

     public static void main(String[] args) {
          System.out.print(function("hello"));
     }

     public static String function(String s){

          if(s.length()==0){
               return "";
          }

          return( s.charAt(s.length()-1) +
                 function(s.substring(0,s.length()-1)) );
     }
}
```

oll eh

4) What is wrong with this code? What error could result? if you skip 1 as an input, infinite recursion.

```
     public static int badMethod(int n){
          if(n==1)
               return 1;
          else
               return n * badMethod(n-2);
     }
```

make it n <= 1

5) Complete the following recursive function:

```
     /*Preconditions: n is a nonnegative integer
      * Postconditions: returns 3^n
      */
     public static int threeRaisedTo(int n){
```

if (n <= 1) return 3;

return 3*threeRaisedTo (n-1)

```
     }
```

t(4)

3 * r(3)

3 * r(2)

3 * r(1)

3

6) Consider the following function:

```
        public static int fun(int b, int e){
               if(e==0)
                      return 1;
               else
                      return b*fun(b, e-1);
        }
```

What is the output of the following?

a) fun(2, 3): 8
b) fun(3, 3): 27
c) fun(2, 4): 16
d) fun(5, 2): 25

7) Write a recursive function that sums all the even numbers less that a given EVEN number n.

8

b 0 e

A) f(2, 3)

4

b e

L 2 * f(2, 2) $\boxed{2}$

b c

2 * f(2, 1)

b e

2 * f(2, 0)

↓

1

b) f(3, 3)

3 * f(3, 2)

3 * f(3, 1)

3 * f(0)

1

7)

```
public static int even(int n) {

    if (n <= 0)
        return 0;

    if (n % 2 == 0)
        return n + even(n-2);
    else
        return even(n-1);

}
```

Java
Tracing Recursion Worksheet #1

Name -
Period -

1.

```
public int sum(int n)
{
   if (n == 1)
      return 1;
   else
      return n + sum(n - 1);
}
```

15

s(5)
|
5 + s(4)
|
4 + s(3)
|
3 + s(2)
|
2 + s(1)
1

What value is returned by the method call `sum(5)` ?

2.

```
public int result(int n)
{
   if (n == 1)
      return 2;
   else
      return 2 * result(n - 1);
}
```

32

r(5)
|
2 * r(4)
|
2 * r(3)
|
2 * r(2)
|
2 * r(1)
2

$2^5 = 32$

What value is returned by the method call `result(5)` ?

3.

```
public int mystery(int n, int a, int d)
{
   if (n == 1)
      return a;
   else
      return d + mystery(n - 1, a, d);
}
```

14

n a d
m(3, 0, 6)
|
6 + m(2, 2, 6)
|
6 + m(1, 2, 6)
2

What value is returned by the method call `mystery(3, 2, 6)` ?

4.

```
public int f(int k, int n)
{
   if (n == k)
      return k;
   else
      if (n > k)
         return f(k, n - k);
      else
         return f(k - n, n);
}
```

$\boxed{2}$

k n
f(6, 8)
|
f(6, 2)
| k n
f(4, 2)
|
f(2, 2)

What value is returned by the method call `f(6, 8)` ?

Java
Tracing Recursion Worksheet #2

Name -
Period -

1.

```
public void strRecur(String s)
{
   if (s.length() < 6)
   {
      System.out.println(s);
      strRecur(s + "*");
   }
}
```

s(wyo) wyo *

s(wyo* wyo *

wyo * *

wyo * **

wyo
wyo*
wyo**

What is displayed by the method call   strRecur("wyo") ?

print ↑
W

2.

```
public void printString(String s)
{
   if (s.length() > 0)
   {
      printString(s.substring(1));
      System.out.print(s.substring(0, 1));
   }
}
```

s(wyo)
s(yo) y
s(o) o

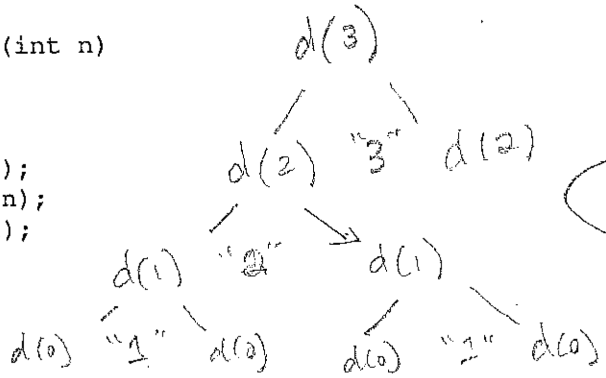
What is displayed by the method call   printString("wyo") ?   o y W

3.

```
public void doSomething(int n)
{
   if (n > 0)
   {
      doSomething(n - 1);
      System.out.print(n);
      doSomething(n - 1);

   }

}
```



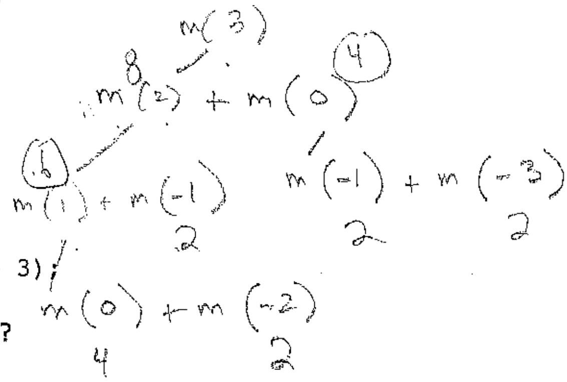
1 2 1 3 1 2 1

What is displayed by the method call   doSomething(3) ?

4.

```
public int mystery(int n)
{

   if (n < 0)
      return 2;
   else
      return mystery(n - 1) + mystery(n - 3);
}
```



What value is returned by the method call   mystery(3) ?

12

Name_______________ Date:__________________ Score:______

AP JAVA

Recursion Worksheet #1

1. For the following method, what would be displayed by the call `mystery1(5)`?

```
public void mystery1(int nNum){
        if(nNum <= 0)
           return;
        else
        {
           System.out.println(nNum);
           mystery1(nNum - 1);
        }
}
```

m(5)
|
m(4)

5
4
3
2
1

2. For the following method, what would be displayed by the call `mystery2(5)`?

```
public void mystery2(int nNum){
      if(nNum <= 0)
           return;
      System.out.println(nNum);
      mystery2(nNum - 1);
}
```

5
4
3
2
1

3. For the following method, what would be displayed by the call: `mystery3(4)`? m(4)

```
public void mystery3(int nNum){
        if(nNum <= 0)
           return;
        for(int nI = 0; nI < nNum; nI++)
           System.out.print("-");
        for(int nI = 0; nI < nNum; nI++)
           System.out.print("+");
        System.out.println();//ends the line
        mystery3(nNum - 1);
}
```

```
----++++
---+++
--++
-+
```

4. For the following method, what value would be returned by the call: ans =`mystery4(4)`?

```
public int mystery4(int nNum){

if  (nNum>1)
        return nNum*mystery4(nNum-2);
else
        return 2;
}
```

m(4)
4 * m(2)
|
2 * m(0)
|
2

16