**Name:**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ SID **#:**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Consider the following recursive definition:

mystery ( 1 ) = 1

mystery ( N ) = mystery ( N – 1 ) + 2 \* N – 1

According to above definition.

* 1. What is mystery ( 5 ) ?
  2. For what set of values N is mystery well defined?
  3. For what set of values N is mystery not defined?
  4. Complete the method below to implement mystery.

public \_\_\_\_\_\_\_\_\_\_\_\_\_ mystery ( \_\_\_\_\_\_\_\_\_\_\_ )

{

if ( \_\_\_\_\_\_\_\_\_\_\_\_ ) //base case;

return \_\_\_\_\_\_\_;

else // general case

{

return \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;

}

}

1. Consider the following recursive definition:

mystery ( 0, Q ) = Q

mystery ( P, Q ) = mystery ( P – 1, Q + 1 )

According to above definition.

* 1. What is mystery ( 2, 4 ) ?
  2. For what set of values P and Q is mystery well defined?
  3. For what set of values P and Q is mystery not defined?
  4. Complete the method below to implement mystery.

public \_\_\_\_\_\_\_\_\_\_\_\_\_ mystery ( \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ )

{

if ( \_\_\_\_\_\_\_\_\_\_\_\_ ) //base case;

return \_\_\_\_\_\_\_;

else // general case

{

return \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;

}

}

1. What does the following method calculate?

public int mystery ( int x, int y )

{

if ( y == 0 ) //base case;

return 0;

else // general case

{

return ( x + mystery(x, y-1) );

}

}

1. Design a recursive method to generate a pattern of stars for a given positive integer number *n*.

For instance, if *n* = 4, shall print:

\*\*\*\*

\*\*\*

\*\*

\*