|  |  |  |
| --- | --- | --- |
| **Data Structures and Algorithms Laboratory** |  | |
| **Laboratory 5:** Recursion | **School of Information Technology** | |
| **Name:** Pattarapon Bunchuai | **ID:** 6431503044 | **Section:** 3 |
| **Date:** | **Due date: on LMS** | |

**Objective**

* To understand the step of recursion
* To create the recursive program
* To implement the factorial and the power program

**Exercise 1:** (in-class) Simple recursive program shows how to trace the recursion.

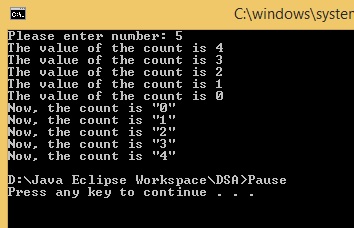
Given the diagram of tracing a countdown algorithm.

input = n

|  |
| --- |
| output |

Create the recursive program to calculate the result of the given diagram. User have to enter the input to the program.

Expected result:



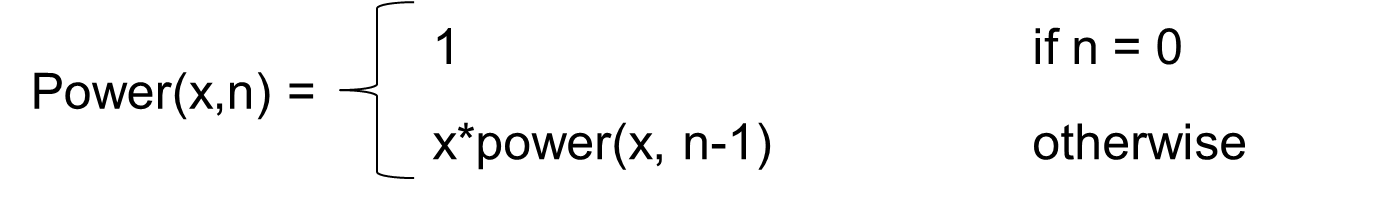
Code (Recursion):

|  |
| --- |
| **public** **static** **void** simRecur(**int** count) {  System.***out***.println("The value of the count is " + --count);  **if** (count > 0) {  *simRecur*(count); //Base case  }  System.***out***.println("Now, the count is " + count);  } |

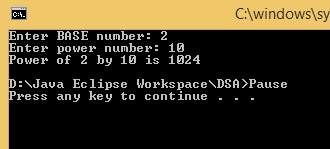
What are the base case and return value of the countdown algorithm?

|  |
| --- |
| When count = 0 |

**Exercise 2:** (in-class) Given the formulation for computing “Power”.



Create the recursive program to calculate the result of “Power” as the expected result.



Code (Recursion):

|  |
| --- |
| **public** **static** **int** powerRecur(**int** x, **int** n) {  **if** (n == 0) {  **return** 1; //Base case  }  **return** x\**powerRecur*(x, n-1);  } |

What are the base case and return value of the Power algorithm?

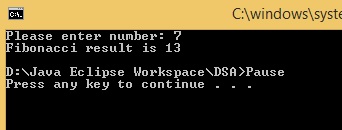
|  |
| --- |
| When n = 0 |

From the same recursive problem, modify the source code by using the repetition LOOP statement to solve the problem.

Code (Loop):

|  |
| --- |
| **public** **static** **int** powerLoop(**int** x, **int** n) {  **int** result = 1;    **for** (**int** i = 1; i <= n; i++) {  result \*= x;  }    **return** result;  } |

**Exercise 3:** (Homework) Create the recursive program to calculate the result of “Fibonacci” as the expected result.



Code (Recursion):

|  |
| --- |
| **public** **static** **int** fibonacci(**int** n) {  **if** (n == 0) {  **return** 0;  }  **if** (n == 1 || n == 2) {  **return** 1; //Real base case  }  **return** *fibonacci*(n - 2) + *fibonacci*(n - 1);  } |

What are the base case and return value of the Fibonacci algorithm?

|  |
| --- |
| When n = 1 or 2 |

From the same recursive problem, modify the source code by using the repetition LOOP statement to solve the problem.

Code (Loop):

|  |
| --- |
| **public** **static** **int** fiboLoop(**int** n) {  **int** current = 0;  **int** next = 1;  **for** (**int** i = 0; i < n; i++) {  **int** sum = current + next;  current = next;  next = sum;  }  **return** current;  } |