HACKERRANK QUESTIONS

DAY - 3

1. import java.io.\*;

import java.util.\*;

public class Solution {

public static void main(String[] args) {

Scanner s = new Scanner(System.in);

int n = s.nextInt();

int temp = n;

int sum = 0;

int rem = 0;

while(n != 0) {

rem = n % 10;

sum += rem;

n = n / 10;

}

if(temp % sum == 0) {

System.out.println("Harshad Number"); }

else {

System.out.println("Not Harshad Number");

}

}

}

2. import java.io.\*;

import java.util.\*;

public class Solution {

public static void main(String[] args) {

Scanner s = new Scanner(System.in);

int n = s.nextInt();

int sum = 0;

int i = 1;

for(i = 1; i < n; i++) {

if(n % i == 0){

sum+=i;

}

}

if(sum > n) {

System.out.println("Abundant Number");

}

else {

System.out.println("Not Abundant Number");

}

}

}

3. import java.util.Scanner;

public class Solution {

public static void main(String[] args) {

Scanner s = new Scanner(System.in);

int n = s.nextInt();

if (n < 100) {

System.out.println("Invalid Input");

return;

}

int sum = 0;

int temp = n;

while (temp != 0) {

sum += temp % 10;

temp /= 10;

}

System.out.println("Sum of digit is " + sum);

}

}

4. import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int n = sc.nextInt();

int m = sc.nextInt();

sc.close();

// Validate constraints

if (n < 1 || n > 20 || m < 1 || m > 20 || n > m) {

System.out.println("Invalid Input");

return;

}

// Use 1-based indexing: F1 = 0, F2 = 1

long[] fib = new long[m + 1]; // indices 0..m, we'll use 1..m

fib[1] = 0;

if (m >= 2) fib[2] = 1;

for (int i = 3; i <= m; i++) {

fib[i] = fib[i - 1] + fib[i - 2];

}

long sum = 0;

for (int i = n; i <= m; i++) {

sum += fib[i];

}

System.out.println("The Sum of Fibonacci value is " + (double) sum);

}

}

5. import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int n = sc.nextInt();

sc.close();

if (n < 1 || n > 9) {

System.out.println("Invalid Input");

return;

}

for (int i = 1; i <= n; i++) {

System.out.println(n + " x " + i + " = " + (n \* i));

}

}

}

6. import java.util.Scanner;

public class Solution {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int n = sc.nextInt();

int m = sc.nextInt();

sc.close();

// Validate constraints and range order

if (n <= 0 || m <= 0 || n >= 30 || m >= 30 || n >= m) {

System.out.println("Invalid Input");

return;

}

int sum = 0;

for (int i = n; i <= m; i++) {

if (i % 2 == 0) {

sum += i;

}

}

System.out.println(sum);

}

}

7. import java.io.\*;

import java.util.\*;

public class Solution {

public static void main(String[] args) {

Scanner s = new Scanner(System.in);

long n = s.nextLong();

if(n>=1 && n<=10000000){

int i=0;

while(n!=0){

n=n/10;

i++;

}

System.out.println("The count of the given integer is: " +i);

} else {

System.out.println("Enter a Valid Input");

}

}

}

8. import java.io.\*;

import java.util.\*;

public class Solution {

public static void main(String[] args) {

Scanner s=new Scanner(System.in);

int n=s.nextInt();

char a='A';

if(n>0&&n<9){

for(int i=0;i<n;i++){

for(int j=0;j<=i;j++){

System.out.print(a+ " ");

a++;

}

System.out.println(" ");

}

}

else{

System.out.println("Invalid Input");

}

}

}

9. import java.util.\*;

public class Solution {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int n = sc.nextInt();

// Print the upper part (including the middle line)

for (int i = 1; i <= n; i++) {

// Print leading spaces

for (int j = 0; j < n - i; j++) {

System.out.print(" ");

}

// Print letters from A to (A + i - 1) separated by spaces

for (int k = 0; k < i; k++) {

System.out.print((char)('A' + k));

if (k < i - 1) System.out.print(" ");

}

System.out.println();

}

// Print the lower part

for (int i = n - 1; i >= 1; i--) {

// Print leading spaces

for (int j = 0; j < n - i; j++) {

System.out.print(" ");

}

// Print letters from A to (A + i - 1)

for (int k = 0; k < i; k++) {

System.out.print((char)('A' + k));

if (k < i - 1) System.out.print(" ");

}

System.out.println();

}

sc.close();

}

}