1. HARSHAD NUMBER 22

import java.util.Scanner;

public class HarshadNumber {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int num = sc.nextInt();

int temp = num;

int digitSum = 0;

while (temp > 0) {

digitSum += temp % 10;

temp /= 10;

}

if (num % digitSum == 0) {

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

} else {

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

}

sc.close();

}

}

2. ABUNDANT NUMBER 11

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int n = sc.nextInt(); // input number

int sum = 0;

// find proper divisors and sum them

for (int i = 1; i <= n / 2; i++) {

if (n % i == 0) {

sum += i;

}

}

if (sum > n) {

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

} else {

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

}

}

}

3.SUM OF DIGIT 10

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int n = sc.nextInt();

if (n <= 99) {

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

} else {

int sum = 0, temp = n;

while (temp > 0) {

sum += temp % 10;

temp /= 10;

}

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

}

}

}

4**.**FIBONACCI SERIES 144

import java.util.Scanner;

public class FibonacciSum {

public static int fibonacci(int n) {

if (n == 1) return 0;

if (n == 2) return 1;

int a = 0, b = 1, c = 0;

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

c = a + b;

a = b;

b = c;

}

return b;

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int start = sc.nextInt();

int end = sc.nextInt();

if (start < 1 || end > 20 || start > end) {

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

} else {

double sum = 0;

for (int i = start; i <= end; i++) {

sum += fibonacci(i); }

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

}

sc.close();

}

}

5.MULTIPLICATION TABLE 79

import java.util.Scanner;

public class MultiplicationTable {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int n = sc.nextInt();

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.SUM OF EVEN NUMBER 1

import java.io.\*;

import java.util.\*;

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();

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

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

} else {

int sumOfEvens = 0;

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

if (i % 2 == 0) {

sumOfEvens += i;

}

}

System.out.println(sumOfEvens);

}

}

}

**7.**ARMSTRONG NUMBER OR NOT 13

import java.util.Scanner;

public class ArmstrongNumberCheck {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

int number = scanner.nextInt();

if (number < 100 || number > 999) {

System.out.println("No");

return;

}

int originalNumber = number;

int sumOfCubes = 0;

while (number > 0) {

int digit = number % 10;

sumOfCubes += digit \* digit \* digit;

number /= 10;

}

if (sumOfCubes == originalNumber) {

System.out.println("Yes");

} else {

System.out.println("No");

}

}

}

8.SWAP 2 DIGIT NUMBER

import java.util.Scanner;

public class SwapTwoDigits {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

String input = scanner.next();

if (input.length() < 2) {

System.out.println("STDOUT");

return;

}

char firstDigit = input.charAt(0);

char secondDigit = input.charAt(1);

int swappedNumber = (secondDigit - '0') \* 10 + (firstDigit - '0');

System.out.println(swappedNumber);

}

}

9.REVERSE A NUMBER USING LOOP 1

import java.util.Scanner;

public class ReverseNumber {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

long n = sc.nextLong();

if (n <= 0 || n > 1000000000) {

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

return;

}

long reverse = 0;

long original = n;

while (n != 0) {

reverse = reverse \* 10 + n % 10;

n /= 10;

}

System.out.println(reverse);

}

}

10.COLLATZ SEQUENCE 7

import java.util.Scanner;

public class CollatzSequence {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.println("Enter a number");

int n = scanner.nextInt();

int count = 0;

System.out.println(n);

while (n != 1) {

if (n % 2 == 0) {

n = n / 2;

} else {

n = 3 \* n + 1;

}

System.out.println(n);

count++;

}

System.out.println("count:" + count);

scanner.close();

}

}

11.COUNT DIGIT IN AN INTEGER 1

import java.util.Scanner;

public class CountDigits {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

long num = sc.nextLong();

if (num < 1 || num > 10000000) {

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

} else {

int count = 0;

long temp = num;

while (temp != 0) {

count++;

temp /= 10;

}

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

}

}

}

12.PRINT A PATTERN4

import java.util.Scanner;

public class AlphabetPattern {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int n = sc.nextInt();

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

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

return;

}

char ch = 'A';

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

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

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

ch++;

}

System.out.println();

}

}

}

13.ALPHABET DIAMOND 1

import java.util.Scanner;

public class AlphabetDiamondPattern {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int n = sc.nextInt();

int totalRows = 2 \* n - 1;

char ch;

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

int currentRow = i <= n ? i : totalRows - i + 1;

for (int space = n - currentRow; space > 0; space--) {

System.out.print(" ");

}

ch = 'A';

for (int col = 1; col <= currentRow; col++) {

System.out.print(ch);

if (col < currentRow) {

System.out.print(" ");

}

ch++;

}

System.out.println();

}

}

}

14.HOLLOW SQUARE PATTERN 4

import java.util.Scanner;

public class HollowSquarePattern {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int n = sc.nextInt();

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

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

if (i == 1 || i == n || j == 1 || j == n) {

System.out.print("\*");

} else {

System.out.print(" ");

}

}

System.out.println();

}

}

}

15.ALPHABET RIGHT TRIANGLE

import java.util.Scanner;

public class AlphabetPattern {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int n = sc.nextInt();

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

char ch = (char) ('A' + i - 1);

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

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

ch++;

}

System.out.println();

}

}

}