**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ**

**Ордена Трудового Красного Знамени**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский технический университет связи и информатики»**

Кафедра «Математическая кибернетика и информационные технологии»

**Отчет по практической работе № 1**

по дисциплине «Информационные технологии и программирование»

Выполнила:

студентка группы БВТ2104

Степанова Дарина Викторовна

Проверила:

Мосева Марина Сергеевна

Москва, 2022

**Цель работы**: необходимо написать 10 функций.

**Задания:**

1. В Java есть единственный оператор, способный обеспечить остаток от операции деления. Два числа передаются в качестве параметров. Первый параметр, разделенный на второй параметр, будет иметь остаток, возможно, ноль. Верните это значение.

Пример:

remainder(1, 3) ➞ 1

remainder(3, 4) ➞ 3

remainder(-9, 45) ➞ -9

remainder(5, 5) ➞ 0

2. Напишите функцию, которая принимает основание и высоту треугольника и возвращает его площадь.

Пример:

triArea(3, 2) ➞ 3

triArea(7, 4) ➞ 14

triArea(10, 10) ➞ 50

3. В этой задаче фермер просит вас сказать ему, сколько ног можно сосчитать среди всех его животных. Фермер разводит три вида:

chickens = 2 legs

cows = 4 legs

pigs = 4 legs

Фермер подсчитал своих животных, и он дает вам промежуточный итог для каждого вида. Вы должны реализовать функцию, которая возвращает общее количество ног всех животных.

Пример:

animals(2, 3, 5) ➞ 36

animals(1, 2, 3) ➞ 22

animals(5, 2, 8) ➞ 50

4. Создайте функцию, которая принимает три аргумента (prob, prize, pay) и возвращает true, если prob \* prize > pay; в противном случае возвращает false. Чтобы проиллюстрировать это: profitableGamble (0,2, 50, 9) должен выдать значение true, поскольку 1 (0,2 \* 50 - 9), а 1> 0.

Пример:

profitableGamble(0.2, 50, 9) ➞ true

profitableGamble(0.9, 1, 2) ➞ false

profitableGamble(0.9, 3, 2) ➞ true

5. Напишите функцию, которая принимает 3 числа и возвращает, что нужно сделать с a и b: они должны быть сложены, вычитаны, умножены или разделены, чтобы получить N. Если ни одна из операций не может дать N, верните "none". 3 числа – это N, a и b.

Пример:

operation(24, 15, 9) ➞ "added"

operation(24, 26, 2) ➞ "subtracted"

operation(15, 11, 11) ➞ "none"

Примечания:

– В качестве тестового ввода используются только целые числа.

– Числа должны быть добавлены, вычтены, разделены или умножены в порядке их появления в параметрах.

6. Создайте функцию, которая возвращает значение ASCII переданного символа.

Пример:

ctoa('A') ➞ 65

ctoa('m') ➞ 109

ctoa('[') ➞ 91

ctoa('\') ➞ 92

7. Напишите функцию, которая берет последнее число из последовательного списка чисел и возвращает сумму всех чисел до него и включая его.

Пример:

addUpTo(3) ➞ 6

// 1 + 2 + 3 = 6

addUpTo(10) ➞ 55

// 1 + 2 + 3 + ... + 10 = 55

addUpTo(7) ➞ 28

// 1 + 2 + 3 + ... + 7 = 28

8. Создайте функцию, которая находит максимальное значение третьего ребра треугольника, где длины сторон являются целыми числами.

Пример:

nextEdge(8, 10) ➞ 17

nextEdge(5, 7) ➞ 11

nextEdge(9, 2) ➞ 10

9. Создайте функцию, которая принимает массив чисел и возвращает сумму его кубов.

Пример:

sumOfCubes([1, 5, 9]) ➞ 855

// Since 1^3 + 5^3 + 9^3 = 1 + 125 + 729 = 855

sumOfCubes([3, 4, 5]) ➞ 216

sumOfCubes([2]) ➞ 8

sumOfCubes([]) ➞ 0

10. Создайте функцию, которая будет для данного a, b, c выполнять следующие действия:

– Добавьте A к себе B раз.

– Проверьте, делится ли результат на C.

Пример:

abcmath(42, 5, 10) ➞ false

// 42+42 = 84,84+84 = 168,168+168 = 336,336+336 = 672, 672+672 = 1344

// 1344 is not divisible by 10

abcmath(5, 2, 1) ➞ true

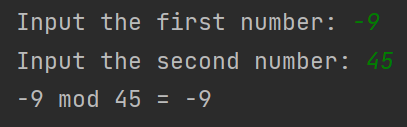
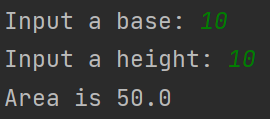
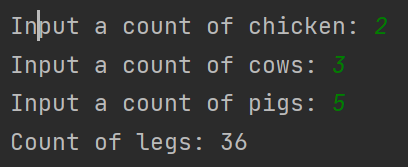
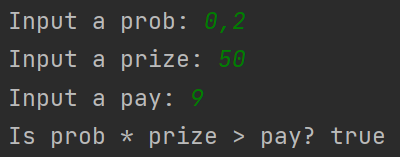
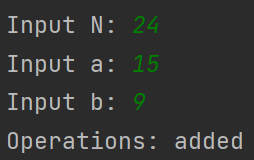
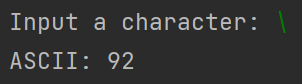
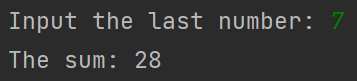
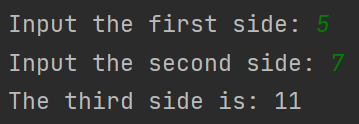
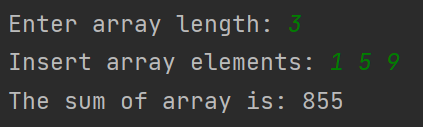
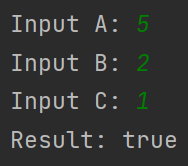
abcmath(1, 2, 3) ➞ false

**Код программы:**

Файл Pract1.java

import java.util.Scanner; // импорт библиотеки для получения данных из консоли  
  
public class Pract1 {  
 public static void main(String[] args) { // вывод в консоль всех функций  
 System.*out*.println(*remainder*());  
 System.*out*.println(*triArea*());  
 System.*out*.println(*animals*());  
 System.*out*.println(*profitableGamble*());  
 System.*out*.println(*operation*());  
 System.*out*.println(*ctoa*());  
 System.*out*.println(*addUpTo*());  
 System.*out*.println(*nextEdge*());  
 System.*out*.println(*sumOfCubes*());  
 System.*out*.println(*abcmath*());  
 }  
 public static int remainder() { // 1. Функция, возвращающая остаток от деления  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.print("Input the first number: ");  
 int f = in.nextInt(); // делимое  
 System.*out*.print("Input the second number: ");  
 int s = in.nextInt(); // делитель  
 System.*out*.print(f + " mod " + s + " = ");  
 return f % s; //остаток от деления  
 }  
 public static double triArea() { // 2. Функция, возвращающая площадь треугольника  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.print("Input a base: ");  
 double base = in.nextInt(); // основание треугольника  
 System.*out*.print("Input a height: ");  
 double height = in.nextInt(); // высота треугольника  
 System.*out*.print("Area is ");  
 return ((base \* height)/2); //площадь  
 }  
 public static int animals() { // 3. Функция, возвращающая общее количество ног животных  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.print("Input a count of chicken: ");  
 int chics = in.nextInt(); //курицы с 2 лапами  
 System.*out*.print("Input a count of cows: ");  
 int cows = in.nextInt(); //коровы с 4 ногами  
 System.*out*.print("Input a count of pigs: ");  
 int pigs = in.nextInt(); //свиньи с 4 ногами  
 System.*out*.print("Count of legs: ");  
 return (chics\*2+cows\*4+pigs\*4); //общее количество ног всех животных  
 }  
 public static boolean profitableGamble() { // 4. Функция, которая принимает три аргумента (prob, prize, pay) и возвращает true, если prob \* prize > pay; в противном случае возвращает false  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.print("Input a prob: ");  
 double prob = in.nextDouble(); //prob  
 System.*out*.print("Input a prize: ");  
 double prize = in.nextDouble(); //prize  
 System.*out*.print("Input a pay: ");  
 double pay = in.nextDouble(); //pay  
 System.*out*.print("Is prob \* prize > pay? ");  
 return prob \* prize > pay; //возвращает true, если prob \* prize > pay, и false в противном случае  
 }  
 public static String operation() { // 5. Функция, которая высчитывает, что нужно сделать с a и b, чтобы получить N  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.print("Input N: ");  
 int n = in.nextInt(); // получаемое число N  
 System.*out*.print("Input a: ");  
 int a = in.nextInt(); // первое число  
 System.*out*.print("Input b: ");  
 int b = in.nextInt(); // второе число  
 String operation = ""; // строка ответа  
 if ((n==a+b)||(n==a\*b)||(n==a/b)||(n==a-b)) { // проверка возможности получить число N четырьмя действиями  
 // добавить к строке возможных действий полученную операцию необходимо, т.к. возможны несколько вариантов  
 // например: 6=6:1=6\*1, 4=2\*2=2+2 и т.д.  
 if (n==a+b) {  
 operation = operation + "added ";  
 }  
 if (n==a\*b) {  
 operation = operation + "multiplicated ";  
 }  
 if (n==a/b) {  
 operation = operation + "divided ";  
 }  
 if (n==a-b) {  
 operation = operation + "subtracted ";  
 }  
 }  
 else {  
 operation = "none"; // ни одна из операций не может дать N  
 }  
 System.*out*.print("Operations: ");  
 return operation; // все действия для получения N  
 }  
 public static int ctoa() { // 6. Функция, которая возвращает значение ASCII переданного символа  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.print("Input a character: ");  
 char character = in.next().charAt(0); // берём первый символ  
 System.*out*.print("ASCII: ");  
 return character; // возвращает int (ASCII)  
 }  
 public static int addUpTo() { // 7. Функция, которая берёт последнее число из последовательного списка чисел и возвращает сумму всех чисел до него и включая его  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.print("Input the last number: ");  
 int n = in.nextInt(); // последнее число  
 int sum = (1 + n)\*n/2; // арифметическая сумма с началом 1  
 System.*out*.print("The sum: ");  
 return sum; // сумма  
 }  
 public static int nextEdge() { // 8. Функция, которая находит максимальное значение третьего ребра треугольника, если длины сторон являются целыми числами  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.print("Input the first side: ");  
 int side1 = in.nextInt(); // одна сторона  
 System.*out*.print("Input the second side: ");  
 int side2 = in.nextInt(); // другая сторона  
 int side3 = side1 + side2 - 1; // сумма двух сторон треугольника должна всегда больше, чем третья сторона, значения целые  
 System.*out*.print("The third side is: ");  
 return side3; // третья сторона  
 }  
 public static int sumOfCubes() { // 9. Функция, которая принимает массив чисел и возвращает сумму его кубов  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.print("Enter array length: ");  
 int size = in.nextInt(); // Читаем с клавиатуры размер массива и записываем в size  
 int[] myArray = new int[size]; // Создаём массив int размером в size  
 System.*out*.print("Insert array elements: ");  
 /\*Пройдёмся по всему массиву, заполняя его\*/  
 for (int i = 0; i < size; i++) {  
 myArray[i] = in.nextInt(); // Заполняем массив элементами, введёнными с клавиатуры  
 }  
 int cubes = 0; // сюда будем прибавлять кубы  
 for (int i = 0; i < size; i++) {  
 cubes += Math.*pow*(myArray[i], 3); // суммируем кубы каждого элемента массива  
 }  
 System.*out*.print("The sum of array is: ");  
 return cubes; // сумма кубов  
 }  
 public static boolean abcmath() { // 10. Функция, которая добавляет себе А по В раз и проверяет, делится ли результат на С  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.print("Input A: ");  
 int a = in.nextInt();  
 System.*out*.print("Input B: ");  
 int b = in.nextInt();  
 System.*out*.print("Input C: ");  
 int c = in.nextInt();  
 System.*out*.print("Result: ");  
 for (int i=0; i<b; i++) {  
 a \*= 2; // дважды берём А, потом берём дважды полученное значение и т.д. В раз  
 }  
 return a % c == 0; // проверка деления результата без остатка  
 }  
}

**Скриншоты выполнения:**

1. 
2. 
3. 
4. 
5. 
6. 
7. 
8. 
9. 
10. 

**Ссылка на GitHub: https://github.com/DarinaStepanova/iteh/**

**Заключение:**

Таким образом, я написала 10 функций для решения некоторых задач.