МИНОБРНАУКИ РОССИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ

УНИВЕРСИТЕТ ИМ. Р.Е. АЛЕКСЕЕВА»

(НГТУ)

Институт радиоэлектроники и информационных технологий

Кафедра «Информационные радиосистемы»

**Контрольная работа по дисциплине  
«Информационные технологии»**

Направление подготовки: \_\_\_\_11.03.01 Радиотехника\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*код и наименование направления подготовки*

Выполнил:

Студент гр. 24-Рз\_\_\_\_\_\_\_\_\_\_\_\_Иванов И.И.

*(группа) (подпись)*

Проверил:

доцент кафедры ИРС \_\_\_\_\_\_\_\_\_\_ Балашова Д.М.

*(подпись)*

Оценка: \_\_\_\_\_\_\_\_\_\_\_\_

Дата: «\_\_\_» \_\_\_\_\_\_\_\_\_\_2024 г.

Нижний Новгород, 2024

**Задание 1**

**Вариант 3**

Найти площадь правильного n-угольника, вписанного в окружность радиуса

R по формуле: . Параметры должны вводиться с клавиатуры.

**Блок-схема алгоритма**



**Листинг программного кода**

#include <stdio.h>

#include <math.h>

int main(){

    int n, R;

    long double S;

    printf("Input n: ");

    scanf("%d", &n);

    printf("Input R: ");

    scanf("%d", &R);

    S = (0.5\*n\*R\*R)\*(sin(2\*M\_PI/n));

    printf("Result is: %Lf\n", S);

    return 0;

}

**Задание 2. Часть 1**

**Вариант 20**

**Блок-схема алгоритма**

**Листинг программного кода**

**Задание 2. Часть 2**

**Вариант 6**

Вычислить элементы векторов  и , , если , , .

**Блок-схема алгоритма**

**Листинг программного кода**

#include <stdio.h>

#include <math.h>

int main(){

    double a = 1000, b = 1;

    double a\_new, b\_new;

    for(int i = 0; i < 6; ++i){

        if (i == 0){

            printf("a = %.2lf, b = %.2lf\n", a, b);

        }

        else{

            a\_new = (a+b)/2;

            b\_new = (sqrt(a\*b));

            a = a\_new, b = b\_new;

            printf("a = %.2lf, b = %.2lf\n", a\_new, b\_new);

        }

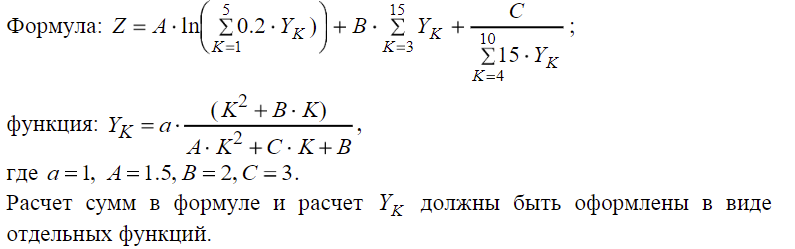
    }

    return 0;

}

**Задание 3**

**Вариант**

**Листинг программного кода**

#include <stdio.h>

#include <math.h>

double calculate\_xi(int i, double a, double b) {

    return b \* (log10(7.2 + i) / 2 + sqrt(i + a + b));

}

double calculate\_sum(int start, int end, double a, double b) {

    double sum = 0;

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

        sum += calculate\_xi(i, a, b);

    }

    return sum;

}

int main() {

    int n = 8;

    double a = 4.0, b = 2.0;

    double sum\_1\_to\_n = calculate\_sum(1, n, a, b);

    double numerator = pow(sum\_1\_to\_n, 2) / n;

    double sum\_3\_to\_n = calculate\_sum(3, n, a, b);

    double second\_term = sum\_3\_to\_n / n;

    double sum\_4\_to\_n = calculate\_sum(4, n, a, b);

    double denominator = sqrt((n - 2) \* sum\_4\_to\_n);

    double r = (numerator - second\_term) / denominator;

    printf("Result (r) = %lf\n", r);

    return 0;

}

**Задание 4 - 1**

**Вариант 9**

Подсчитать количество слов в строке, которая не может начинаться с пробелов, но может заканчиваться ими. Учесть наличие двух и более пробелов между словами (два и более пробелов считать за один).

**Листинг программного кода**

**Задание 4 - 2**

**Вариант**

**Листинг программного кода**

**Задание 5 - 1**

**Вариант**

**Листинг программного кода**

**Задание 5 - 2**

**Вариант**

**Листинг программного кода**