Министерство цифрового развития, связи и массовых коммуникаций

Российской Федерации

Ордена Трудового Красного Знамени

федеральное государственное бюджетное образовательное учреждение

высшего образования

Московский технический университет связи и информатики

(МТУСИ)

Кафедра «Информационная безопасность»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №3

по дисциплине

«Программирование в системах информационной безопасности»

на тему

«Регулярные и итерационные циклы»

Вариант №2

Выполнил:

студент группы БСУ1801 Гавриков А.Г.

Проверил:

старший преподаватель кафедры ИБ

Барков В. В.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Москва 2021

**Цель**

Овладеть навыками создания циклических алгоритмов на языке C с применением инструкций цикла на примере алгоритмов работы с последовательностями чисел

Вариант №2



Рисунок 1 – Исходная формула

**Цикл for**

double summ(int n)

{

    int i;

    double sum;

    sum = 0;

    for(i = 0; i <= n - 1; i++)

    {

        sum +=  pow(-1, i)\*((i+1)/(i\*i\*i+2));

    }

    return (sum);

}

Листинг 1. 1.c

#include "ft.h"

double summ2(double eps)

{

    double sum;

    int i;

    double a;

    i = 0;

    a =  pow(-1, i)\*((i+1)/(i\*i\*i+2));

    sum = 0;

    for (i = 0; fabs(a) > eps; i++)

    {

        sum += a;

        a =  pow(-1, i)\*((i+1)/(i\*i\*i+2));

    }

    return (sum);

}

Листинг 2. 2.c

#include "ft.h"

void print(int n, int k)

{

    int i;

    double z;

    z = 0;

    for (i = 0; i <= n - 1; i++)

    {

        z =  pow(-1, i)\*((i+1)/(i\*i\*i+2));

        if (i % k == 0)

        {

            continue ;

        }

        printf("%.4lf ", z);

    }

}

Листинг 3. 3.c

#include "ft.h"

int findFirstElement(double eps)

{

    double z;

    int i;

    i = 0;

    z =  pow(-1, i)\*((i+1)/(i\*i\*i+2));

    for (i = 0; ; i++)

    {

        z =  pow(-1, i)\*((i+1)/(i\*i\*i+2));

        if (fabs(z) < eps)

        {

            break ;

        }

    }

    return (i);

}

Листинг 4. 4.c

#include "ft.h"

int findFirstNegativeElement(double eps)

{

    int i;

    double z;

    i = 0;

    z =  pow(-1, i)\*((i+1)/(i\*i\*i+2));

    for (i = 0; ; i++)

    {

        z =  pow(-1, i)\*((i+1)/(i\*i\*i+2));

        if (fabs(z) > eps && z < 0)

        {

            return (i);

        }

    }

}

Листинг 5. 5.c

#ifndef FT\_H

# define FT\_H

#include <stdio.h>

#include <math.h>

double summ(int n);

double summ2(double eps);

void print(int n, int k);

int findFirstElement(double eps);

int findFirstNegativeElement(double eps);

#endif

Листинг 6. ft.c

#include "ft.h"

#include <stdlib.h>

int main()

{

    int operator;

    int n = 0;

    int k = 0;

    double eps = 0;

    printf("Task with for\n");

    printf("1. Task 1\n");

    printf("2. Task 2\n");

    printf("3. Task 3\n");

    printf("4. Task 4\n");

    printf("5. Task 5\n");

    printf("6. Exit\n");

    printf("Select number of task: ");

    scanf("%d", &operator);

    switch (operator)

    {

            case 1:

            {

            printf("Enter n: ");

            scanf("%d", &n);

            printf("%.4lf", summ(n));

            break ;

            }

            case 2:

            {

            printf("Enter eps: ");

            scanf("%lf", &eps);

            printf("%.4lf", summ2(eps));

            break ;

            }

            case 3:

            {

            printf("Enter n: ");

            scanf("%d", &n);

            printf("Enter k: ");

            scanf("%d", &k);

            print(n, k);

            break ;

            }

            case 4:

            {

            printf("Enter eps: ");

            scanf("%lf", &eps);

            printf("%d", findFirstElement(eps));

            break ;

            }

            case 5:

            {

            printf("Enter eps: ");

            scanf("%lf", &eps);

            printf("%d", findFirstNegativeElement(eps));

            break ;

            }

            case 6:

            break ;

    }

    system("pause");

}

Листинг 7. main.c

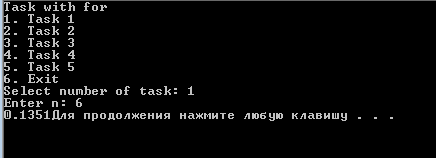


Рисунок 1. Результат работы for

**Цикл while**

#include "ft.h"

double summ(int n)

{

    int i;

    double sum;

    i = 0;

    sum = 0;

    while (i <= n - 1)

    {

        sum += pow(-1, i)\*((i+1)/(i\*i\*i+2));

        i++;

    }

    return (sum);

}

Листинг 8. 1.c

#include "ft.h"

double summ2(double eps)

{

    double sum;

    int i;

    double a;

    i = 0;

    a = pow(-1, i)\*((i+1)/(i\*i\*i+2));

    sum = 0;

    while (fabs(a) > eps)

    {

        sum += a;

        a = pow(-1, i)\*((i+1)/(i\*i\*i+2));

        i++;

    }

    return (sum);

}

Листинг 9. 2.c

#include "ft.h"

void print(int n, int k)

{

    int i;

    double z;

    i = 0;

    z = 0;

    while(i <= n - 1)

    {

        z = pow(-1, i)\*((i+1)/(i\*i\*i+2));

        if (i % k == 0)

        {

            i++;

            continue;

        }

        printf("%.4lf ", z);

        i++;

    }

}

Листинг 10. 3.c

#include "ft.h"

int findFirstElement(double eps)

{

    double z;

    int i;

    i = 0;

    z = pow(-1, i)\*((i+1)/(i\*i\*i+2));

    while (fabs(z) > eps)

    {

        z = pow(-1, i)\*((i+1)/(i\*i\*i+2));

        if (fabs(z) < eps)

        {

            break;

        }

        i++;

    }

    return (i);

}

Листинг 11. 4.c

#include "ft.h"

int findFirstNegativeElement(double eps)

{

    int i;

    double z;

    i = 0;

    z = pow(-1, i)\*((i+1)/(i\*i\*i+2));

    while (fabs(z) > eps)

    {

        z = pow(-1, i)\*((i+1)/(i\*i\*i+2));

        if (z < 0)

        {

            return (i);

        }

        i++;

    }

    return (0);

}

Листинг 12. 5.c

#ifndef FT\_H

# define FT\_H

#include <stdio.h>

#include <math.h>

double summ(int n);

double summ2(double eps);

void print(int n, int k);

int findFirstElement(double eps);

int findFirstNegativeElement(double eps);

#endif

Листинг 13. ft.c

#include "ft.h"

#include <stdlib.h>

int main()

{

    int operator;

    int n = 0;

    int k = 0;

    double eps = 0;

    printf("Task with while\n");

    printf("1. Task 1\n");

    printf("2. Task 2\n");

    printf("3. Task 3\n");

    printf("4. Task 4\n");

    printf("5. Task 5\n");

    printf("6. Exit\n");

    printf("Select number of task: ");

    scanf("%d", &operator);

    switch (operator)

    {

            case 1:

            {

            printf("Enter n: ");

            scanf("%d", &n);

            printf("%.4lf", summ(n));

            break ;

            }

            case 2:

            {

            printf("Enter eps: ");

            scanf("%lf", &eps);

            printf("%.4lf", summ2(eps));

            break ;

            }

            case 3:

            {

            printf("Enter n: ");

            scanf("%d", &n);

            printf("Enter k: ");

            scanf("%d", &k);

            print(n, k);

            break ;

            }

            case 4:

            {

            printf("Enter eps: ");

            scanf("%lf", &eps);

            printf("%d", findFirstElement(eps));

            break ;

            }

            case 5:

            {

            printf("Enter eps: ");

            scanf("%lf", &eps);

            printf("%d", findFirstNegativeElement(eps));

            break ;

            }

            case 6:

            break ;

    }

    system("pause");

}

Листинг 14. main.c

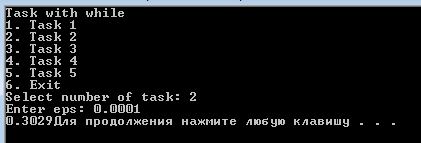


Рисунок 2. Результат работы while

**Цикл do…while**

#include "ft.h"

double summ(int n)

{

    int i;

    double sum;

    i = 0;

    sum = 0;

    do

    {

        sum += pow(-1, i)\*((i+1)/(i\*i\*i+2));

        i++;

    } while (i < n);

    return (sum);

}

Листинг 15. 1.c

#include "ft.h"

double summ2(double eps)

{

    double sum;

    int i;

    double a;

    i = 0;

    a = pow(-1, i)\*((i+1)/(i\*i\*i+2));

    sum = 0;

    do

    {

        sum += a;

        a = pow(-1, i)\*((i+1)/(i\*i\*i+2));

        i++;

    } while (fabs(sum) > eps);

    return (sum);

}

Листинг 16. 2.c

#include "ft.h"

void print(int n, int k)

{

    int i;

    double z;

    i = 0;

    z = 0;

    do

    {

        z = pow(-1, i)\*((i+1)/(i\*i\*i+2));

        if ((i + 1) % k == 0)

        {

            i++;

            continue ;

        }

        printf("%.4lf ", z);

        i++;

    } while(i < n);

}

Листинг 17. 3.c

#include "ft.h"

int findFirstElement(double eps)

{

    double z;

    int i;

    i = 0;

    z = pow(-1, i)\*((i+1)/(i\*i\*i+2));

    do

    {

        z = pow(-1, i)\*((i+1)/(i\*i\*i+2));

        if (fabs(z) < eps)

        {

            break ;

        }

        i++;

    } while (fabs(z) > eps);

    return (i);

}

Листинг 18. 4.c

#include "ft.h"

int findFirstNegativeElement(double eps)

{

    int i;

    double z;

    i = 0;

    z = pow(-1, i)\*((i+1)/(i\*i\*i+2));

    do

    {

        z = pow(-1, i)\*((i+1)/(i\*i\*i+2));

        if (z < 0)

        {

            return (i);

        }

        i++;

    } while (fabs(z) > eps);

    return (0);

}

Листинг 19. 5.c

#ifndef FT\_H

# define FT\_H

#include <stdio.h>

#include <math.h>

double summ(int n);

double summ2(double eps);

void print(int n, int k);

int findFirstElement(double eps);

int findFirstNegativeElement(double eps);

#endif

Листинг 20. ft.c

#include "ft.h"

#include <stdlib.h>

int main()

{

    int operator;

    int n = 0;

    int k = 0;

    double eps = 0;

    printf("Task with do ... while\n");

    printf("1. Task 1\n");

    printf("2. Task 2\n");

    printf("3. Task 3\n");

    printf("4. Task 4\n");

    printf("5. Task 5\n");

    printf("6. Exit\n");

    printf("Select number of task: ");

    scanf("%d", &operator);

    switch (operator)

    {

            case 1:

            {

            printf("Enter n: ");

            scanf("%d", &n);

            printf("%.4lf", summ(n));

            break ;

            }

            case 2:

            {

            printf("Enter eps: ");

            scanf("%lf", &eps);

            printf("%.4lf", summ2(eps));

            break ;

            }

            case 3:

            {

            printf("Enter n: ");

            scanf("%d", &n);

            printf("Enter k: ");

            scanf("%d", &k);

            print(n, k);

            break ;

            }

            case 4:

            {

            printf("Enter eps: ");

            scanf("%lf", &eps);

            printf("%d", findFirstElement(eps));

            break ;

            }

            case 5:

            {

            printf("Enter eps: ");

            scanf("%lf", &eps);

            printf("%d", findFirstNegativeElement(eps));

            break ;

            }

            case 6:

            break ;

    }

    system("pause");

}

Листинг 21. main.c

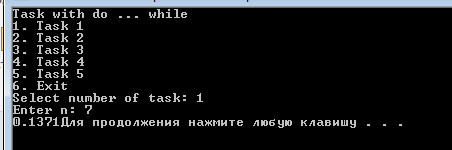


Рисунок 3. Результат работы do…while

**Вывод**

В данной работе были разобраны аспекты работы с функциями for, while и do while. Так же были разобраны операторы continue и break.