ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ  
Ордена Трудового Красного Знамени федеральное государственное   
бюджетное учреждение высшего образования

Московский технический университет связи и информатики

МТУСИ

Кафедра информационной безопасности

Лабораторная работа

Вариант 2

Выполнил студент группы БИН1703 ОТФ-2 \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Бен Режеб Софиэн

Старший преподаватель кафедры ИТ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В. Барков

**Практикум №3**  
**Регулярные и итерационные циклы**

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

***Задание 1***

Написать функцию для вычисления выражения .

Функция в качестве параметра принимает значение и вычисляет сумму слагаемых.

Прототип функции double summ(int n). Функция не должна использовать функции консольного ввода-вывода.

***Задание 2***

Написать функцию для вычисления выражения с точностью .

Функция в качестве параметра принимает значение и вычисляет сумму до тех пор, пока не выполнится условие .

Прототип функции double summ2(double eps). Функция не должна использовать функции консольного ввода-вывода.

***Задание 3***

Написать функцию, которая печатает членов последовательности , исключая из неё каждый -ый член. Числа и передаются в функцию в качестве параметра. Для исключения члена последовательности используйте инструкцию перехода continue.

Прототип функции void print(int n, int k).

***Задание 4***

Написать функцию, которая возвращает номер первого члена последовательности , для которого выполняется условие .

Функция в качестве параметра принимает значение .

Выход из цикла организовать с помощью инструкции перехода break.

Прототип функции int findFirstElement(double eps).

Функция не должна использовать функции консольного ввода-вывода.

***Задание 5***

Написать функцию, которая возвращает номер первого отрицательного члена последовательности , для которого выполняется условие .

Функция в качестве параметра принимает значение .

Выход из цикла организовать с помощью инструкции перехода return.

Прототип функции int findFirstNegativeElement(double eps).

Функция не должна использовать функции консольного ввода-вывода.

***Задание 6***

Разработать функцию main(), которая в цикле выводит на экран меню из 6 пунктов и ожидает ответа пользователя:

1. Задание 1
2. Задание 2
3. Задание 3
4. Задание 4
5. Задание 5
6. Выход

Обработку пользовательского ввода выполнить с помощью инструкции switch. При нажатии на соответствующую цифру меню стирается с экрана и приложение переходит в режим демонстрации работы выбранного задания. По завершении демонстрации программа ожидает пользовательского ввода для перехода в меню. Выход из программы должно осуществляться при нажатии цифры 6 в режиме меню. Скомпилировать 3 версии программы:

1. Все функции реализованы с помощью цикла for
2. Все функции реализованы с помощью цикла while
3. Все функции реализованы с помощью цикла do … while

Убедиться, что все три версии программы работают одинаково

**Индивидуальные варианты заданий**

2

Листинг 1 — Исходный файл header\_do.h

#ifndef header\_h

#define header\_h

#include <stdio.h>

#include <math.h>

double a ; double eps; int n; int k; double s;

double sum(int n);

double sum2(double eps);

void print(int n, int k);

int LetsFindFirstElement(double eps);

int LetsFindFirstNegativeElement(double eps);

#endif

Листинг 2 — Исходный файл main\_do.c

#include <stdio.h>

#include <math.h>

#include <locale.h>

#include "header\_do.h"

int main()

{double r;

printf("Lets chose number of task:\n1-Task1.\n2-Task2.\n3-Task3.\n4-Task4.\n5-Task5.\n");

switch(getchar())

{case '1':

printf("Task1:\n");

printf("Input n:\n");

scanf("%i", &n);

printf("n=%i\n", n);

r=sum(n);

printf("sum=%1f/n",r);

break;

case '2':

printf("Task2:\n");

printf("Input eps:\n");

scanf("%lf", &eps);

printf("eps=%lf\n", eps);

r=sum2(eps);

printf("sum2=%1f/n",r);

break;

case '3':

printf("Task3:\n");

printf("Input n:\n");

scanf("%i", &n);

printf("Input k:\n");

scanf("%i", &k);

print(n, k);

break;

case '4':

printf("Task4:\n");

printf("Input eps:\n");

scanf("%lf", &eps);

printf("eps=%lf\n", eps);

r=LetsFindFirstElement(eps);

printf("First Element=%1f/n",r);

break;

case '5':

printf("Task5:\n");

printf("Input eps:\n");

scanf("%lf", &eps);

printf("eps=%lf\n", eps);

r=LetsFindFirstNegativeElement(eps);

printf("First Negative Element=%1f/n",r);

break;

default:

printf("Unavalible var\n");

break;}

}

Листинг 3 — Исходный файл task1\_do.c

#include <stdio.h>

#include <math.h>

#include <locale.h>

#include "header\_do.h"

double sum(int n)

{

a=0;

s = 0.0;

int i = 0;

while (i < n)

{

a=pow(-1,i)\*(1)/((double)((i+1)\*(i+2)\*(i+3)));

s =s+ a;

++i;

}

return (s);

}

Листинг 4 — Исходный файл task2\_do.c

#include <stdio.h>

#include <math.h>

#include <locale.h>

#include "header\_do.h"

double sum2(double eps)

{

a = eps + 1.0;

s = 0.0;

int i = 0;

while (fabs(a) > eps)

{

a=pow(-1,i)\*(1)/((double)((i+1)\*(i+2)\*(i+3)));

s += a;

++i;

}

return (s);

}

Листинг 5 — Исходный файл task3\_do.c

#include <stdio.h>

#include <math.h>

#include <locale.h>

#include "header\_do.h"

void print(int n, int k)

{

{

int i = 0;

while (i < n)

{

a=pow(-1,i)\*(1)/((double)((i+1)\*(i+2)\*(i+3)));

if ((i + 1) % k == 0)

{

++i;

continue;

}

++i;

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

}

}

}

Листинг 6 — Исходный файл task4\_do.c

#include <stdio.h>

#include <math.h>

#include <locale.h>

#include "header\_do.h"

int LetsFindFirstElement(double eps)

{

int i = 0;

n=-1;

while (i > -1)

{

a=pow(-1,i)\*(1)/((double)((i+1)\*(i+2)\*(i+3)));

if (fabs(a) <= eps)

{

n = i + 1;

break;

}

++i;

}

return(n);

}

Листинг 7 — Исходный файл task5\_do.c

#include <stdio.h>

#include <math.h>

#include <locale.h>

#include "header\_do.h"

int LetsFindFirstNegativeElement(double eps)

{

int i = 0;

while (i > -1)

{

a=pow(-1,i)\*(1)/((double)((i+1)\*(i+2)\*(i+3)));

if (fabs(a) <= eps && a < 0.0)

{ ++i;

}

}

return(i + 1);

}

Листинг 8 — Исходный файл header\_do\_while.h

#ifndef header\_h

#define header\_h

#include <stdio.h>

#include <math.h>

double a ; double eps; int n; int k; double s;

double sum(int n);

double sum2(double eps);

void print(int n, int k);

int LetsFindFirstElement(double eps);

int LetsFindFirstNegativeElement(double eps);

#endif

Листинг 9 — Исходный файл task1\_do\_while.c

#include <math.h>

#include <stdio.h>

#include <locale.h>

#include "header.h"

double sum(int n)

{

a=0;

int i = 0;

do

{

a=pow(-1,i)\*(1)/((double)((i+1)\*(i+2)\*(i+3)));

++i;

} while (i < n - 1);

return a;

}

Листинг 10 — Исходный файл task2\_do\_while..c

#include <math.h>

#include <stdio.h>

#include <locale.h>

#include "header.h"

double sum2(double eps)

{

a=0;

n=1;

int i = 0;

do

{

a=a+pow(-1,i)\*(1)/((double)((i+1)\*(i+2)\*(i+3)));

if (fabs(a) <= eps)

n = n - 1;

else

++n;

++i;

} while (i < n - 1);

return a;

}

Листинг 11 — Исходный файл task3\_do\_while.c

#include <math.h>

#include <stdio.h>

#include <locale.h>

#include "header.h"

void print(int n, int k) {

int i = 0;

do

{

if (i == k)

{

++i;

continue;

}

a = pow(-1,i)\*(1)/((double)((i+1)\*(i+2)\*(i+3)));

printf("%f\n", a);

++i;

} while (i < n - 1);

}

Листинг 12 — Исходный файл main\_do\_while.c

#include <stdio.h>

#include <math.h>

#include <locale.h>

#include "header.h"

int main()

{

double r;

printf("Lets chose number of task:\n1-Task1.\n2-Task2.\n3-Task3.\n4-Task4.\n5-Task5.\n");

switch(getchar())

{

case '1':

printf("Task1:\n");

printf("Input n:\n");

scanf("%i", &n);

printf("n=%i\n", n);

r=sum(n);

printf("sum=%1f/n",r);

break;

case '2':

printf("Task2:\n");

printf("Input eps:\n");

scanf("%lf", &eps);

printf("eps=%lf\n", eps);

r=sum2(eps);

printf("sum2=%1f/n",r);

break;

case '3':

printf("Task3:\n");

printf("Input n:\n");

scanf("%i", &n);

printf("Input k:\n");

scanf("%i", &k);

print(n, k);

break;

case '4':

printf("Task4:\n");

printf("Input eps:\n");

scanf("%lf", &eps);

printf("eps=%lf\n", eps);

r=LetsFindFirstElement(eps);

printf("First Element=%1f/n",r);

break;

case '5':

printf("Task5:\n");

printf("Input eps:\n");

scanf("%lf", &eps);

printf("eps=%lf\n", eps);

r=LetsFindFirstNegativeElement(eps);

printf("First Negative Element=%1f/n",r);

break;

default:

printf("Unavalible var\n");

break;

}

}

Листинг 13 — Исходный файл task4\_do\_while.c

#include <math.h>

#include <stdio.h>

#include <locale.h>

#include "header.h"

int LetsFindFirstElement(double eps)

{

n = 1;

int j;

int i = 0;

do

{

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

j = i;

if (fabs(a) <= eps)

{

break;

}

else

{

++n;

}

++i;

} while (i < n - 1);

return j;

}

Листинг 14 — Исходный файл task5\_do.c

#include <stdio.h>

#include <math.h>

#include <locale.h>

#include "header\_do.h"

int LetsFindFirstNegativeElement(double eps)

{

int i = 0;

while (i > -1)

{

a=pow(-1,i)\*(1)/((double)((i+1)\*(i+2)\*(i+3)));

if (fabs(a) <= eps && a < 0.0)

{ ++i;

}

}

return(i + 1);

}

Листинг 15 — Исходный файл main\_for.c

#include <stdio.h>

#include <math.h>

#include <locale.h>

#include "header\_for.h"

int main()

{

double r;

printf("Lets chose number of task:\n1-Task1.\n2-Task2.\n3-Task3.\n4-Task4.\n5-Task5.\n");

switch(getchar())

{

case '1':

printf("Task1:\n");

printf("Input n:\n");

scanf("%i", &n);

printf("n=%i\n", n);

r=sum(n);

printf("sum=%1f/n",r);

break;

case '2':

printf("Task2:\n");

printf("Input eps:\n");

scanf("%lf", &eps);

printf("eps=%lf\n", eps);

r=sum2(eps);

printf("sum2=%1f/n",r);

break;

case '3':

printf("Task3:\n");

printf("Input n:\n");

scanf("%i", &n);

printf("Input k:\n");

scanf("%i", &k);

print(n, k);

break;

case '4':

printf("Task4:\n");

printf("Input eps:\n");

scanf("%lf", &eps);

printf("eps=%lf\n", eps);

r=LetsFindFirstElement(eps);

printf("First Element=%1f/n",r);

break;

case '5':

printf("Task5:\n");

printf("Input eps:\n");

scanf("%lf", &eps);

printf("eps=%lf\n", eps);

r=LetsFindFirstNegativeElement(eps);

printf("First Negative Element=%1f/n",r);

break;

default:

printf("Unavalible var\n");

break;

}

}

Листинг 16 — Исходный файл task1\_for.c

#include <stdio.h>

#include <math.h>

#include <locale.h>

#include "header\_for.h"

double sum(int n)

{

a = 0;

s = 0.0;

int i = 0;

for(i=0;i<=n-1;i++)

{

a =pow(-1,i)\*(1)/((double)((i+1)\*(i+2)\*(i+3)));

s = s + a;

}

return (s);

}

Листинг 17 — Исходный файл task2\_for.c

#include <stdio.h>

#include <math.h>

#include <locale.h>

#include "header\_for.h"

double sum2(double eps)

{

a = eps + 1.0;

s = 0.0;

int i = 0;

for (i = 0; fabs(a) <= eps; ++i) {

a = pow(-1,i)\*(1)/((double)((i+1)\*(i+2)\*(i+3)));

s += a;

}

return (s);

}

Листинг 18 — Исходный файл task3\_for.c

#include <stdio.h>

#include <math.h>

#include <locale.h>

#include "header\_for.h"

void print(int n, int k)

{

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

{

a = pow(-1,i)\*(1)/((double)((i+1)\*(i+2)\*(i+3)));

if ((i + 1) % k == 0)

{

continue;

}

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

}

}

Листинг 19 — Исходный файл task4\_for.c

#include <stdio.h>

#include <math.h>

#include <locale.h>

#include "header\_for.h"

int LetsFindFirstElement(double eps)

{

n=-1;

for (int i = 0; ; ++i)

{

a = pow(-1,i)\*(1)/((double)((i+1)\*(i+2)\*(i+3)));

if (fabs(a) <= eps)

{

n = i + 1;

break;

}

}

return(n);

}

Листинг 20 — Исходный файл task5\_for.c

#include <stdio.h>

#include <math.h>

#include <locale.h>

#include "header\_for.h"

int LetsFindFirstNegativeElement(double eps)

{

for (int i = 0; ; ++i)

{

a = pow(-1,i)\*(1)/((double)((i+1)\*(i+2)\*(i+3)));

if (fabs(a) <= eps && a < 0)

{

return(i + 1);

}

}

}

Листинг 21 — Исходный файл header\_for.h

#ifndef header\_for\_h

#define header\_for\_h

#include <math.h>

#include <stdio.h>

double a ; double eps; int n; int k; double s;

double sum(int n);

double sum2(double eps);

void print(int n, int k);

int LetsFindFirstElement(double eps);

int LetsFindFirstNegativeElement(double eps);

#endif

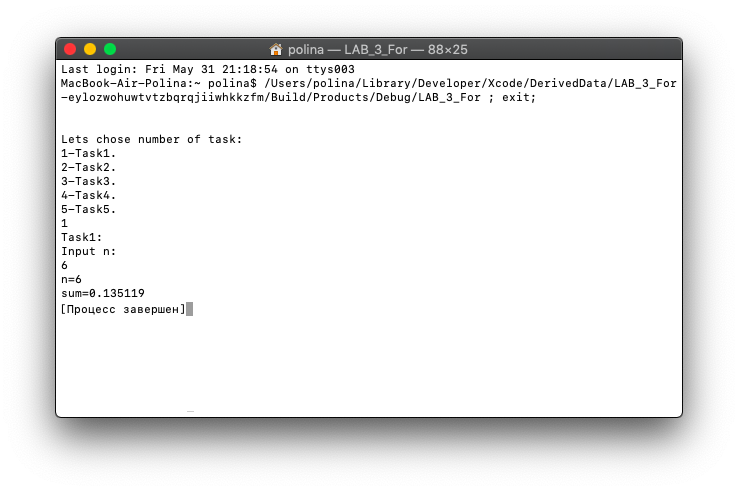


Рисунок 1



Рисунок 2

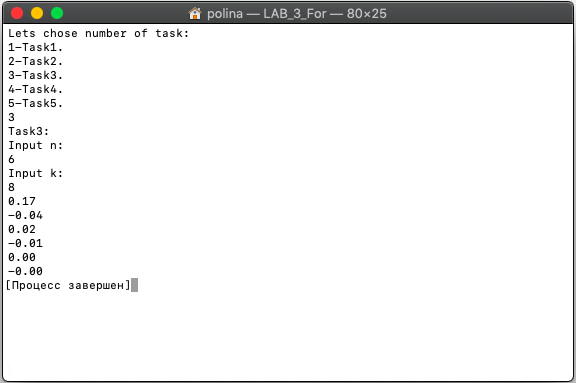


Рисунок 3

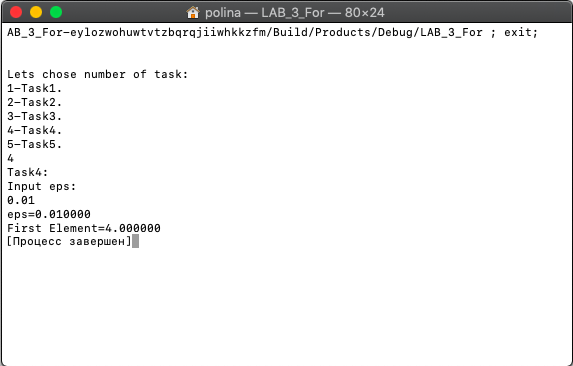


Рисунок 4

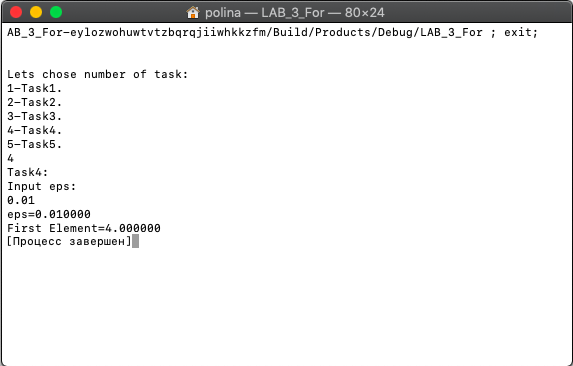


Рисунок 5