# Лабораторная работа 8

“Функции в языке СИ”

## Задание 5

Познакомьтесь с синтаксисом и семантикой функции rand().

Запустите несколько раз код на выполнение. Удостоверьтесь, что

при вызове функции rand() генерируется последовательность,

повторяющаяся при каждом новом выполнении программы.

Измените код, выполните рандомизацию (функцию srand() без

использования ―семени‖), чтобы достичь эффекта генерации

различных последовательностей случайных чисел для каждого

выполнения программы.

## Код программы

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

#include<locale.h>

int main()

{

srand(time(NULL));

char\*locale = setlocale(LC\_ALL, "");

int i;

for (i=1;i<=20;i++)

{

printf("%3d",1+(rand()%6));

if (i%5==0)

printf("\n");

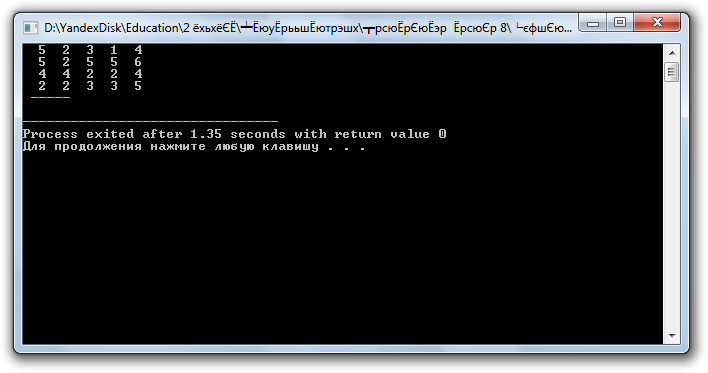
}

printf(" ————-\n");

getch();

}

## Результат проделанной работы



## Задание 4

Ознакомьтесь с программой.

Ответьте на вопросы:

1) за что отвечает библиотека assert?

2) какое математическое действие выполняет функция StepFast?

## Ответы на вопросы

1. **assert.h** — [заголовочный файл](https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%B3%D0%BE%D0%BB%D0%BE%D0%B2%D0%BE%D1%87%D0%BD%D1%8B%D0%B9_%D1%84%D0%B0%D0%B9%D0%BB" \o "Заголовочный файл) [стандартной библиотеки](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D0%BD%D0%B0%D1%8F_%D0%B1%D0%B8%D0%B1%D0%BB%D0%B8%D0%BE%D1%82%D0%B5%D0%BA%D0%B0" \o "Стандартная библиотека) [языка программирования С](https://ru.wikipedia.org/wiki/C_(%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F)" \o "C (язык программирования)), в котором объявляется макрос препроцессора языка С assert(). Данный макрос реализует исключение, которое может использоваться для проверки сделанных программой вычислений. В данной программе проверяет правильность вычисления.
2. StepFast вычисляет степень числа.

## Код программы

#include<stdio.h>

#include<conio.h>

#include <locale.h>

#include<assert.h>

int StepFast (int x,int y) {

int a=x,b=y,z=1;

while (b!=0)

if (odd(b)) {

z\*=a; b--;

}

else {

a\*=a; b/=2;

}

return z;

}

int odd (int t) {

return (t%2==0)?0:1;

}

void main() {

char \*locale = setlocale(LC\_ALL, "");

printf("Тестирование...\n");

assert(StepFast(5,3)==125);

assert(StepFast(11,4)==121\*121);

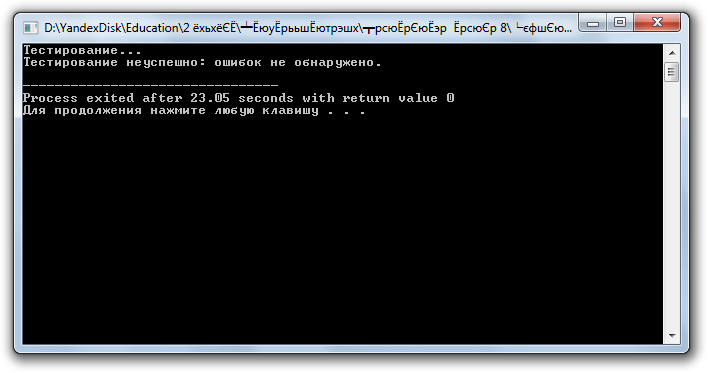
printf("Тестирование неуспешно: ошибок не обнаружено.\n");

getch();

return 0;

}

## Результат проделанной работы



## Задание 3

Допишите программу таким образом, чтобы она соответствовала

предметной области и результату.

Предметная область

Функция main() вызывает функцию more(), а функция more()

вызывает функцию main(), т.е. организован цикл с помощью

взаимного вызова функций

Результат работы программы:

Введите символ (Q - признак окончания): 3

Вы указали символ 3

Введите символ (Q - признак окончания): С

Вы указали символ С

Введите символ (Q - признак окончания): Q

Вы указали символ Q

## Код программы

#include<stdio.h>

#include<conio.h>

#include <locale.h>

void more (char x) {

if(x!='Q') {

main();

}

}

int main() {

char \*locale = setlocale(LC\_ALL, "");

char ch;

printf("Введите символ (Q - признак окончания): ");

scanf("%c",&ch);

fflush(stdin); /\* Очистка буфера клавиатуры \*/

printf("Вы указали символ %c\n",ch);

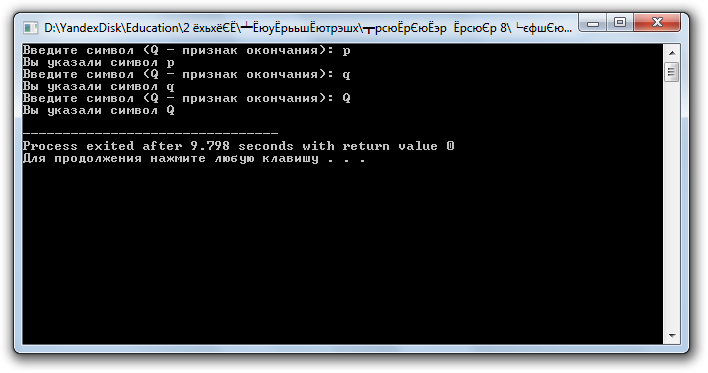
more(ch);

getch();

return 0;

}

## Результат проделанной работы



## Задание 2

Познакомьтесь с синтаксисом и семантикой функции rand().

Запустите несколько раз код на выполнение. Удостоверьтесь, что

при вызове функции rand() генерируется последовательность,

повторяющаяся при каждом новом выполнении программы.

Измените код, выполните рандомизацию (функцию srand() без

использования ―семени‖), чтобы достичь эффекта генерации

различных последовательностей случайных чисел для каждого

выполнения программы.

## Код программы

#include<stdio.h>

#include<conio.h>

#include <locale.h>

int func() {

printf("Мы устали!\n");

}

int main() {

char \*locale = setlocale(LC\_ALL, "");

int i;

for (i=1;i<=3;i++)

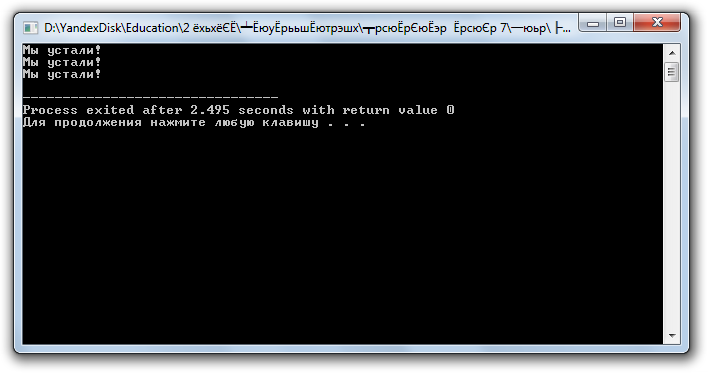
func();

getch();

return 0;

}

## Результат проделанной работы



## Задание 1

Запустите программу и ответьте на вопросы, заключенные в

комментариях.

## Код программы

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

#include <locale.h>

int main()

{

char \*locale = setlocale(LC\_ALL, "");

double z1,z2;

printf("Введите числа z1 и z2: ");

scanf("%le %le",&z1,&z2);

printf(" ceil(%g)=%le\n",z1,ceil(z1)); /\* Функция ceil выполняет округление и возвращает ближайшее целое значение к заданному числу,

но это значение будет не меньше самого \*/

printf(" floor(%g)=%le\n",z1,floor(z1)); /\* Функция floor выполняет округление заданного значения и возвращает наибольшее целое значение,

которое не больше, чем заданное значение\*/

printf(" fmod(%g,%g)=%le\n",z1,z2,fmod(z1,z2)); /\* Функция fmod вычисляет остаток от деления и возвращает значение с плавающей точкой,

оставшуюся часть от целочисленного деления параметров z1 и z2\*/

printf(" sin(%g)=%g\n",z1,sin(z1));

printf(" cos(%g)=%g\n",z1,cos(z1));

printf(" tan(%g)=%g\n",z1,tan(z1));

printf(" sinh(%g)=%g\n",z1,sinh(z1)); // Гиперболический синус от z1

printf(" cosh(%g)=%g\n",z1,cosh(z1)); // Гиперболический косинус от z1

printf(" tanh(%g)=%g\n",z1,tanh(z1)); // Гиперболический тангенс от z1

getch();

return 0;

}

## Результат проделанной работы

