**Лабораторная работа № 8.**

**Функции в языке С.**

1. Постановка задачи

Код программы

Результаты выполненной работы

2. Постановка задачи

Допишите код в указанных местах.

Объясните свой выбор

1) типа возвращаемого значения функции

2) списка параметров функции

Код программы

#include<stdio.h>

#include<conio.h>

/\* ———— \*/

int func() /\* Вызываемая функция \*/

{

printf("Мы устали!\n");

}

int main()

{

int i;

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

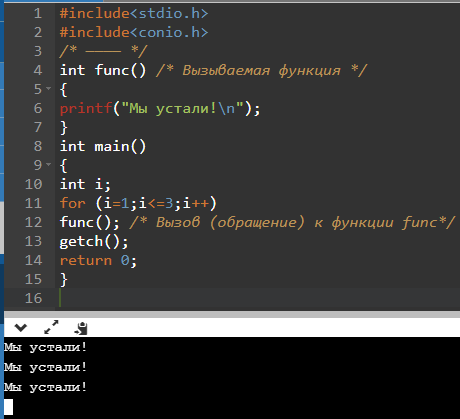
func(); /\* Вызов (обращение) к функции func\*/

getch();

return 0;

}

Результаты выполненной работы



3. Постановка задачи

Допишите программу таким образом, чтобы она соответствовала предметной области и результату.

Код программы

#include<stdio.h>

#include<conio.h>

int more(char c)

{

if (c!='Q')

main(c);

}

int main()

{

char ch;

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

scanf("%c",&ch);

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

printf("Вы указали символ %c\n",ch);

/\* В этой части необходимо прописать блок кода,

чтобы было соответствие с результатом и ПО\*/

more(ch);

getch();

return 0;

}

/\* Вызываемая функция \*/

Результаты выполненной работы

4. Постановка задачи

Ознакомьтесь с программой.

Ответьте на вопросы:

1) за что отвечает библиотека assert?

2) какое математическое действие выполняет функция StepFast?

Код программы

#include<stdio.h>

#include<conio.h>

#include<assert.h>

int main()

{

printf("Тестирование...\n");

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

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

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

getch();

return 0;

}

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;

}

Результаты выполненной работы

1) за что отвечает библиотека assert?

Библиотека assert отвечает за вывод ошибок и за завершение работы программы

2) какое математическое действие выполняет функция StepFast?

функция StepFast проверяет на нечетность, функция возвращает значение z.

5. Постановка задачи

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 <time.h>

/\* ————– \*/

int main()

{

int i;

srand(time(NULL));

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

{

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

if (i%5==0)

printf("\n");

}

printf(" ————-\n");

getch();

return 0;

}

Результаты выполненной работы

