## Практические задания для Урока 3 Модуля 2.

1. Напишите программу для вычисления значения функции F(n), где n – натуральное число, задана следующими соотношениями:

F(1) = 3

F(n) = F(n–1) \* (n–1), при n >1

Пользователь вводит n, программа должна вывести значение F(n).

1. Напишите программу для вычисления значения функции F(n), где n – натуральное число, задана следующими соотношениями:

F(1) = 1

F(n) = F(n–1) \* F(n–1) − F(n–1) \* n + 2 \* n, при n >1

Пользователь вводит n, программа должна вывести значение F(n).

1. Напишите программу для вычисления значения функции F(n) и G(m), где n и m – натуральные числа, задана следующими соотношениями:

F(1) = 0

F(n) = F(n–1) + n, при n >1

G(1) = 1

G(n) = G(n–1) \* n, при n >1

Пользователь вводит n и m, программа должна вывести F(n)+G(m).

1. Напишите программу для вычисления значения функции F(n), где n – натуральное число, задана следующими соотношениями:

F(1) = 2

F(2) = 5

F(n) = F(n–1) \* n + F(n–2) \* (n – 1) , при n >2

Пользователь вводит n, программа должна вывести значение F(n).

1. Напишите программу для вычисления значения функции F(n), где n – натуральное число, задан следующими соотношениями:

F(1) = 1

F(2) = 3

F(n) = F(n−1) \* F(n−2) + (n−2), при n > 2

Пользователь вводит n, программа должна вывести значение F(n).

1. Напишите программу для вычисления значения функции Аккермана для двух неотрицательных целых чисел *n* и *m*, где

A(n,m)= m+1, если n=0

A(n,m)= A(n-1,1), если n<>0,m=0

A(n,m)=A(n-1, A(n,m-1)) , если n>0 и m>0

Пользователь вводит n и m, программа должна вывести значение F(n,m).

## Решения.

1.

#include <stdio.h>

int F(int n)

{

if (n == 1)

return 3;

if (n > 1)

return F(n-1) \* (n - 1);

return 0;

}

int main()

{

int n;

printf("n= ");

scanf("%d", &n);

printf("%d\n",F(n));

return 0;

}

2.

#include <stdio.h>

int F(int n)

{

if (n == 1)

return 1;

if (n > 1)

return F(n-1) \* F(n-1) - F(n-1) \* n + 2 \* n;

return 0;

}

int main()

{

int n;

printf("n= ");

scanf("%d", &n);

printf("%d\n",F(n));

return 0;

}

3.

#include <stdio.h>

int F(int n)

{

if (n == 1)

return 0;

if (n > 1)

return F(n-1) + n;

return 0;

}

int G(int n)

{

if (n == 1)

return 1;

if (n > 1)

return G(n - 1) \* n;

return 0;

}

int main()

{

int n,m;

printf("n= ");

scanf("%d", &n);

printf("m= ");

scanf("%d", &m);

printf("%d\n",F(n)+G(m));

return 0;

}

4.

#include <stdio.h>

int F(int n)

{

if (n == 1)

return 2;

if (n == 2)

return 5;

if (n > 2)

return F(n-1) \* n + F(n-2) \* (n-1);

return 0;

}

int main()

{

int n;

printf("n= ");

scanf("%d", &n);

printf("%d\n",F(n));

return 0;

}

5.

#include <stdio.h>

int F(int n)

{

if (n == 1)

return 1;

if (n == 2)

return 3;

if (n > 2)

return F(n-1) \* F(n - 2) + n-2;

return 0;

}

int main()

{

int n;

printf("n= ");

scanf("%d", &n);

printf("%d\n",F(n));

return 0;

}

6.

#include <stdio.h>

unsigned int A(unsigned int n, unsigned int m)

{

if (n==0)

return m+1;

else

if ((n!=0) && (m==0))

return A(n-1,1);

else

return A(n-1,A(n,m-1));

}

int main()

{

int n,m;

printf("n=");

scanf("%d",&n);

printf("m=");

scanf("%d",&m);

printf("%d\n",A(n,m));

return 0;

}