Описание спецификаций функций

Спецификация функции LogFunc2

1.Заголовок: void LogFunc2(int \*a)

2.Назначение:Вычисляет значение таблицы истинности для выражения 2 описанного выше и записывает его в а

Спецификация функции LogFunc4

1.Заголовок: void LogFunc4(int \*a)

2.Назначение:Вычисляет значение таблицы истинности для выражения 4 описанного выше и записывает его в а

Спецификация функции PrintTable

1.Заголовок: void PrintTable(int \*a,int \*b)

2.Назначение: Выводит таблицу истинности для a и b, которые вычисляются в LogFunc2 и LogFunc4

Функция CheckEqualBool описывается в лабораторной работе 1.1

#include <stdio.h>

#include <locale.h>

int CheckEqualBool(int\* a,int\* b)

{

int i=0;

int f=1;

while((i<8)&&f){

f=(a[i]==b[i]);

i++;

}

return f;

}

void LogFunc2(int \*a)

{

for (int i=0;i<8;i++)

{

int A=(1<<2&i)>>2;

int B=(1<<1&i)>>1;

int C=1&i;

a[i]=(A&B&~C)|(~A&C);

}

}

void LogFunc4(int \*a)

{

for (int i=0;i<8;i++)

{

int A=(1<<2&i)>>2;

int B=(1<<1&i)>>1;

int C=1&i;

a[i]=(A&B&~C)|(~A&C);

}

}

void PrintTable(int \*a,int \*b)

{

printf("A B C 2 4\n");

for(int i=0;i<8;i++)

printf("%d %d %d %d %d\n",((1<<2&i)>>2),((1<<1&i)>>1),(1&i),a[i],b[i]);

}

int main()

{

setlocale(LC\_ALL,"Rus");

int a[8],b[8];

LogFunc2(a);

LogFunc4(b);

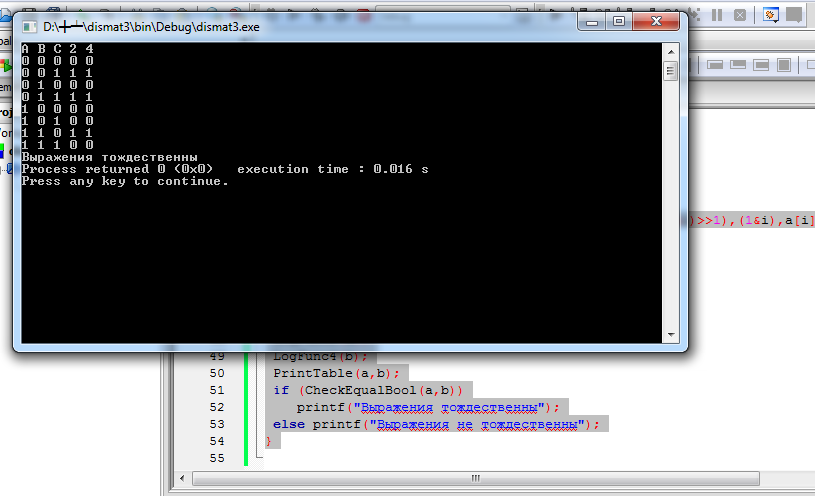
PrintTable(a,b);

if (CheckEqualBool(a,b))

printf("Выражения тождественны");

else printf("Выражения не тождественны");

}



Спецификация функции Func2

1.Заголовок:

void Func3(int \*A,int \*B,int \*C, int \*Rez)

2.Назначение: Вычисляет значение выражения 2 описанного выше и записывает его в Rez

Спецификация функции Func4

1.Заголовок:

void Func4(int \*A,int \*B,int \*C,int \*Rez)

2.Назначение:Вычисляет значение выражения 4 описанного выше и записывает его в Rez

Функции для работы со множествами, хранящимися в виде массива Булевского типа, используемые в функциях main, Func2, Func4 описаны в лабораторной работе 1.1

#include <stdio.h>

#include <locale.h>

void GetAddBool(int\* a,int\* c)

{

int i;

for (i=0;i<7;i++)

c[i]=(!a[i]);

}

void PrintSetBool(int\* a)

{

int i;

for (i=0;i<7;i++)

if (a[i])

printf("%d ",i+1);

printf("\n");

}

int CheckEqualBool(int\* a,int\* b)

{

int i=0,f=1;

while((i<7)&&f)

{

f=(a[i]==b[i]);

i++;

}

return f;

}

void GetDiffBool(int\* a,int\* b,int\* c)

{

for(int i=0;i<7;i++)

c[i]=a[i]>b[i];

}

void GetSumBool(int\* a,int\* b,int\* c)

{

for (int i=0;i<7;i++)

c[i]=(a[i]||b[i]);

}

void GetCrossBool(int\* a,int\* b,int\* c)

{

for (int i=0;i<7;i++)

c[i]=(a[i]&&b[i]);

}

void Func3(int \*A,int \*B,int \*C, int \*Rez)

{

int Rez1[7],Rez2[7];

GetCrossBool(C,A,Rez);

GetDiffBool(C,Rez,Rez);

GetCrossBool(A,B,Rez1);

GetCrossBool(Rez1,C,Rez2);

GetDiffBool(Rez1,Rez2,Rez1);

GetSumBool(Rez,Rez1,Rez);

}

void Func4(int \*A,int \*B,int \*C,int \*Rez)

{

int addA[7],addB[7],addC[7],Rez1[7],Rez2[7];

GetAddBool(A,addA);

GetAddBool(B,addB);

GetAddBool(C,addC);

GetSumBool(addA,addC,Rez);

GetAddBool(Rez,Rez);

GetDiffBool(C,Rez,Rez);

GetSumBool(addA,addB,Rez1);

GetSumBool(Rez1,addC,Rez2);

GetAddBool(Rez1,Rez1);

GetAddBool(Rez2,Rez2);

GetDiffBool(Rez1,Rez2,Rez1);

GetSumBool(Rez,Rez1,Rez);

}

int main()

{

setlocale(LC\_ALL,"Rus");

int A[7]={1,0,1,0,1,0,1},B[7]={0,1,1,0,0,1,1},C[7]={0,0,0,1,1,1,1};

int Rez3[7],Rez4[7];

Func3(A,B,C,Rez3);

Func4(A,B,C,Rez4);

if (CheckEqualBool(Rez3,Rez4))

{

printf("Выражения тождественны \n");

PrintSetBool(Rez3);

}

else

{

printf("Выражения нетождественны \n");

PrintSetBool(Rez3);

PrintSetBool(Rez4);

}

}

