**Лабораторная работа № 9.**

**Операции над указателями.**

1. Постановка задачи

Объявите две переменные целого типа. На каждую из них ссылается указатель. На первую ссылается указатель p\_1, а на

вторую указатель p\_2. Кроме того, объявлена переменная

типа double и указатель на неѐ p\_dbl.

Используя указатели, подсчитать частное целых переменных

(первую делим на вторую) и сохранить это значение в переменную,

на которую ссылается p\_dbl.

Код программы

#include <stdio.h>

int main()

{

int x,y,\*p\_1,\*p\_2;

double z,\*p\_dbl;

scanf("%d %d",&x,&y);

p\_1=&x;

p\_2=&y;

p\_dbl=&z;

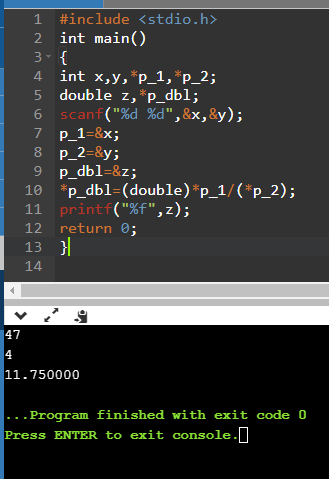
\*p\_dbl=(double)\*p\_1/(\*p\_2);

printf("%f",z);

return 0;

}

Результаты выполненной работы



2. Постановка задачи

Что выполняет приведенная программа? Найдите ошибку в

программе и объясните ее причину. Исправьте программу так,

чтобы она работала корректно.

Код программы

#include <stdio.h>

void swap(int \*a, int \*b) {

int temp = \*a;

\*a = \*b;

\*b = temp;

}

int main(){

int x = 3,y = 5;

printf("x=%d \t y=%d \n",x,y);

swap(&x,&y);

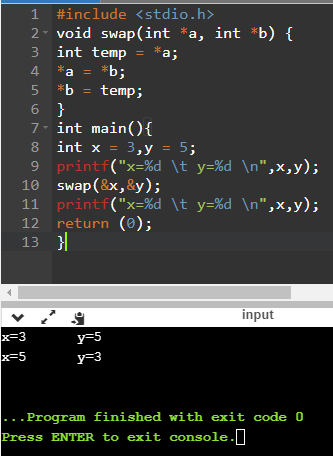
printf("x=%d \t y=%d \n",x,y);

return (0);

}

Результаты выполненной работы

Программа меняет местами значения переменных. Программа работала некорректно из-за того, что при завершении функции, копии переменных удаляются.



3. Постановка задачи

Что выполняется в данной программе? Дайте ответ и напишите

комментарии к программе.

Результаты выполненной работы

Программа меняет местами значения переменных, с помощью оператора sizeof и работы с байтами.

#include <conio.h>

#include <conio.h>

#include <stdio.h>

void main() {

int length;

char \*p1, \*p2;

char tmp;

float a = 5.0f;

float b = 3.0f;

printf("a = %.3f\n", a);

printf("b = %.3f\n", b);

p1 = (char\*) &a;

p2 = (char\*) &b;

length = sizeof(float); // Сколько байт необходимо поменять

while (length--) {

tmp = \*p1;

\*p1 = \*p2; // Меняем местами значения переменных побайтно

\*p2 = tmp;

p1++;

p2++;

}

printf("a = %.3f\n", a);

printf("b = %.3f\n", b);

getch();

}

4. Постановка задачи

Допустимо ли в Си? Если "да" - опишите семантику каждого правильного действия (не принимая во внимание ошибочные); если "нет" - объясните почему.

Код программы

Результаты выполненной работы

a).

int i, ∗ p, j, ∗q; /\*да, присвоение типа int переменным\*/

p = &i; /\* да, указатель 'p' указывает на переменную 'i' \*/ q = &p; /\* нет, разные типы данных\*/

j = ∗p = 1; /\* да, переменные i и j получают значение 1\*/ q = p-1; /\* нет, тк это указатели\*/ ∗p += 1; /\*да, переменная i увеличивается на 1\*/

i = ∗++q + ∗p; q -= 1; i = ∗q ++ + ∗q; /\* нет, тк значение q не указано\*/

printf("i=%d, j=%d, ∗p=%d, ∗q=%d \n", i, j, ∗p, ∗q); /\*да, вывод переменных\*/

b).

int x = 1, y; /\*да, присвоение типа int переменным\*/

char c = ‘a’; int ∗pi, ∗qi; char ∗pc;

pi = &x;/\*да, присвоение указателю переменной\*/ ∗pi = 3; y = ∗pi; ∗pi = c; qi = pi; /\*да, обмен переменными\*/

pc = qi; ∗qi+=1; pi++; ∗(- - pi) = 5; y = ∗qi+1;

pc = &c; ++\*pc; (∗pc)++; ∗pc++; ∗pc+=1;

x = (int)pi; pi=(int∗)pc; pi=(int∗)x; x = 1+ ∗pi; pc=(char∗)pi;

c = ∗pc; pc = &y; x = qi – pi; qi = 0; qi+=pi;

y = &pi; y = (int)&pi; pi = pi +5; ∗(pi+1)=0; pi=&(x+0);