Лабораторная работа №9.

Операции с указателями.

Задание 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 неё p_dbl.
Используя указатели, подсчитать частное целых переменных (первую делим на вторую) и сохранить это значение в переменную, на которую ссылается p_dbl.

Код:

```c
#include <stdio.h>

int main(void) {
int a = 10;
int b = 3;
int *p_1 = &a;
int *p_2 = &b;
double chastnoe;
double *p_dbl = &chastnoe;
*p_dbl = *p_1 / *p_2;
printf("a / b = %lf",chastnoe);
return 0;
}
```

Результат выполнения:

```
a / b = 3.000000
```

Задание 2:

Постановка задачи:

Что выполняет приведенная программа? Найдите ошибку в программе и объясните ее причину. Исправьте программу так, чтобы она работала корректно.

Код:

```c
#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);
}
//Ошибка была в том что функция должна принимать в себя указатели, а не значения.
```

Результат выполнения:

```
x=3       y=5
x=5       y=3
```

Задание 3:

Постановка задачи:

Что выполняется в данной программе? Дайте ответ и напишите комментарии к программе.

Код:

```
#include <conio.h>
#include <conio.h>
#include <stdio.h>
void main() {
int length; //Обьявление переменно1 length
char *p1, *p2; //Обьявление указателей
char tmp;
float a = 5.0f;
float b = 3.0f;

printf("a = %.3f\n", a);//Вывод на экран значений a и b
printf("b = %.3f\n", b);
p1 = (char*) &a;//  Присваиваем указателю p1 адрес переменной a но в типе char
p2 = (char*) &b;//  Присваиваем указателю p2 адрес переменной b но в типе char
length = sizeof(float);
while (length--) {//Цикл с управляющей переменной length, выход из которого выполняется при значении аргумента равным нулю
tmp = *p1;//Обмен значений переменных
*p1 = *p2;
*p2 = tmp;
p1++;//Переходы к следующим адресам
p2++;
}
printf("a = %.3f\n", a);
printf("b = %.3f\n", b);
getch();
}
```

Результат выполнения:

```
a = 5.000
b = 3.000
a = 3.000
b = 5.000
```

Задание 4:

Постановка задачи:

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

Код:

```c
#include <conio.h>
#include <conio.h>
#include <stdio.h>
void main() {
int i, * p, j, *q;
p = &i;//Присваивание адреса переменной i и указателю p;
q = &p;//Присваивание адреса указателя p и указателю q;
j = *p = 1; //Присвоение j значения i = 1;
q = p-1; // Присвоение значения указателя на переменную перд i;
*p += 1; //Увелечение значения переменной хранящейся по адресу в указатели на 1
i = *++q + *p; //Присваиваем i значение переменной следующий за Переменной q
q -= 1; // Указатель на переменную перед q
i = *q ++ + *q;//Присваиваем i
printf("i=%d, j=%d, *p=%d, *q=%d \n", i, j, *p, *q);//Вывод значений переменных
}
```

```c
#include <conio.h>
#include <conio.h>
#include <stdio.h>
void main()
{
int x = 1, y;
char c = 'a';
int *pi, *qi;
char *pc;
pi = &x; //Присваиваем адрес переменной x указателю pi
*pi = 3; //Присваиваем переменной находящей по адресу хранящемся в pi значение 3
 y = *pi; //Присваиваем y значение переменной находящей по адресу хранящемся в pi
*pi = c; //Присваиваем переменной находящей по адресу хранящемся в pi значение c
qi = pi; //Присваиваем указателю qi адрес находящийся в Pi
pc = qi; //Не верное действие, тк указателю на char присваивают значение указателя int
*qi+=1; //Увелечение значения переменной находящейся по адресу хранящемся в qi
pi++;//Перемещение указателя на следующую переменную
 *(--pi) = 5; //Присвоить указателю значения адреса предшесвтующей переменной и присвоить ей значение 5
 y = *qi+1; //Присваиваем y значение переменной по адресу в qi + 1
pc = &c; //присваиваем указателю адрес c
++*pc;//Получение значение переменной следующей за той, что указана в pc и увелеченеи на 1
 (*pc)++;//Увелечение значения на 1
  *pc++; //Получение значения следующей переменной
  *pc+=1; //Увелечение значения на 1
x = (int)pi; //Неверное действие тк пропущена звездочка
pi=(int*)pc;//Перевод адреса из типа char в int
pi=(int*)x;//Неверное действие - попытка присвоить указателю значение
x = 1+ *pi; //Увелечение значения на 1
pc=(char*)pi;//Перевод адреса из типа int в char
c = *pc; //Присвоение пересенной с значения еременной находящей по адресу хранящемся в pc
pc = &y; //передача адреса y в pc Неверное действие тк попытка передать адрес переменной типа int указателю char
x = qi - pi; //Неверное действие вычитание адресов
qi = 0; //Обнуление указателя
qi+=pi;//Неверное
y = &pi;
y = (int)&pi;
pi = pi +5;
*(pi+1)=0;
pi=&(x+0);
}
```