**Самостоятельная работа № 7.**

**Указатели на простейшие типы. Операции над указателями.**

1. Постановка задачи

Каждой строке кода сопоставьте графическое представление.

Код программы

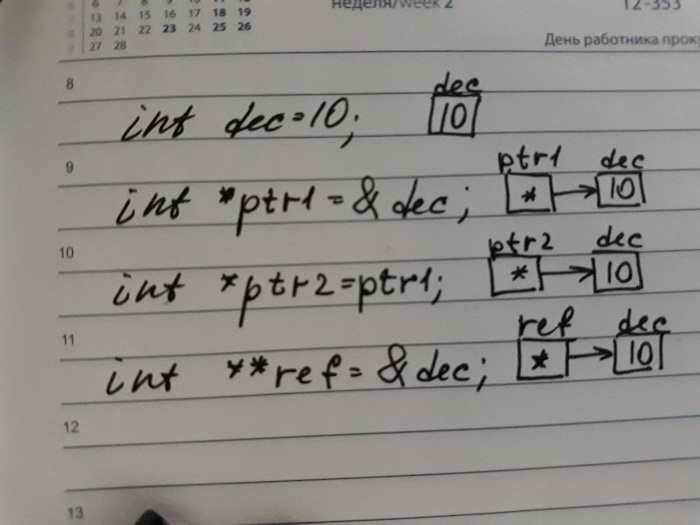
int dec=10;

int \*ptr1=&dec;

int \*ptr2=ptr1;

int \*\*ref=&dec;

Результаты выполненной работы



2. Постановка задачи

Дано следующее описание переменных:

int \*p,\*q;

char \*r;

Какие из следующих операторов присваивания неправильны и

почему?

(а) p=q; (в) p=NULL; (д) q=\*p;

(б) q=r; (г) r=NULL; (е) \*p=NULL.

Результаты выполненной работы

(а) p=q; //правильно

(б) q=r; //неправильно, так как у переменных разные типы

(в) p=NULL; //правильно

(г) r=NULL; //правильно

(д) q=\*p; //неправильно, тк это два указателя

(е) \*p=NULL. //неправильно, тк перед р не ставится \*

3. Постановка задачи

Дано следующее описание переменных:

int \*p,\*q;

char \*r;

Какие из следующих операторов неправильны и почему?

(а) \*r=\*p; (б) \*q=(int)\*r;

(в) if (r!=NULL) (г) if (q>NULL)

\*r=\*NULL; \*q=\*p;

(д) if (q==p) (e) if (q!=r)

printf("%d",q); scanf("%c",r);

Результаты выполненной работы

(а) \*r=\*p; //неправильно, так как у переменных разные типы

(б) \*q=(int)\*r; //правильно

(в) if (r!=NULL)

\*r=\*NULL; //NULL не является указателем

(г) if (q>NULL)

\*q=\*p; //неправильно, тк с NULL сравнивается не значение по адресу q, а адрес записанный в q с NULL

(д) if (q==p)

printf("%d",q); //правильно

(e) if (q!=r)

scanf("%c",r); //неправильно, переменные нельзя сравнивать, тк у них разные типы

4. Постановка задачи

Выполните каждое из следующих заданий, используя для этой

цели только один оператор. Считайте объявленными переменные

value1 и value2 типа long, причѐм переменной value11 присвоено

значение 20000.

(а) Объявите указатель lPtr на объект данных типа long.

(б) Присвойте значение адреса переменной value1 указателю lPtr.

(в) Выведите значение объекта, на который ссылается lPtr.

(г) Присвойте значение объекта, на который ссылается lPtr, переменной

value2.

(д) Выведите значение value2.

(е) Выведите адрес value1.

(ж) Выведите значение адреса, находящееся в lPtr.

Математическая модель

Список идентификаторов

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| volue1 | Объявленная переменная | long |
| volue2 | Переменная, которой мы присваиваем значение указателя | long |
| lPtr | Указатель | long |

Код программы

#include <stdio.h>

int main()

{

long volue1=20000,value2;

long \*lPtr;

lPtr=&volue1;

printf("Значение объекта, на который ссылается 1Ptr = %d\n",\*lPtr);

value2=\*lPtr;

printf("значение value2 = %d\n",value2);

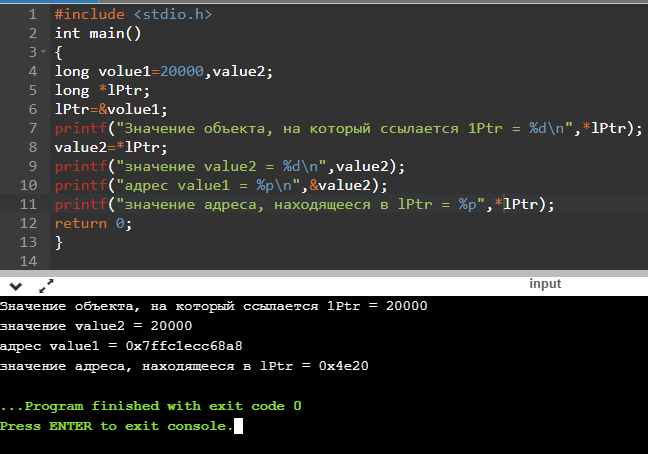
printf("адрес value1 = %p\n",&value2);

printf("значение адреса, находящееся в lPtr = %p",\*lPtr);

return 0;

}

Результаты выполненной работы



5. Постановка задачи

Найдите ошибку в каждом из следующих программных

фрагментов. Если ошибку можно исправить, пропишите, как это

сделать.

Код программы

(а) int \*number;

printf("%d\n",\*number);

number не инициализирован

(б) float \*realPtr;

long \*integerPtr;

integerPtr=realPtr;

Мы не можем приравнивать разные типы данных

(в) int \*x,y;

x=y;

после знака равно необходимо поставить & (x=&y)

(г) short \*numPtr, result;

void \*genericPtr=numPtr;

result=\*genericPtr+7;

Необходимо убрать \* (result=genericPtr+7;)

(д) float x=19.34;

float xPtr=&x;

printf("%f\n",xPtr);

Исправлено:

float x=19.34;

float \*xPtr=&x;

printf("%f\n",\*xPtr);

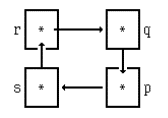
(е) char \*s;

printf("%s\n",s);

s не инициализирован

6. Постановка задачи

Напишите программный фрагмент, который представлен графически следующим образом:



Код программы

Int \*\*r,\*\*s=&r,\*\*p=&s,\*\*q=&p, \*\*q=&r;