

ЛАБОРАТОРНАЯ РАБОТА № 18

ПРОГРАММИРОВАНИЕ С ИСПОЛЬЗОВАНИЕМ УКАЗАТЕЛЕЙ

Цель работы: приобрести практические навыки при использовании техники указателей.

Краткие теоретические сведения

Указатель – это переменная, которая может содержать адрес некоторого объекта в памяти компьютера, например адрес другой переменной. И через указатель, установленный на переменную можно обращаться к участку оперативной памяти, отведенной компилятором под ее значения.

Указатель объявляется следующим образом:

тип *идентификатор;

Примеры описаний указателей:

int *pti; char *ptc; float *ptf;

После такого описания переменная pti может принимать значение указателя на величину целого типа; переменная ptc предназначена для хранения указателя на величину типа char; переменная ptf – на величину типа float.

С указателями связаны две унарные операции: & и *. Операция & означает «взять адрес» операнда (т.е. установить указатель на операнд). Данная операция допустима только над переменными. Операция * имеет смысл: «значение, расположенное по указанному адресу» и работает следующим образом:

- определяется местоположение в оперативной памяти переменной типа указатель.

- извлекается информация из этого участка памяти и трактуется как адрес переменной с типом, указанным в объявлении указателя.

- производится обращение к участку памяти по выделенному адресу для проведения некоторых действий.

Пример 1:

```
int x,          /* переменная типа int */
    *y;         /* указатель на элемент данных типа int */
y=&x;           /* y - адрес переменной x */
*y=1;           /* косвенная адресация указателем поля x
                "по указанному адресу записать 1", т.е. x=1; */
```

Пример 2:

```
int i, j=8, k=5, *y;
y=&i;
*y=2;           /* i=2 */
y=&j; /* переустановили указатель на переменную j */
*y+=i;          /* j+=i , т.е. j=j+1 -> j=j+2=10 */
y=&k; /* переустановили указатель на переменную k */
k+=*y;          /* k+=k, k=k+k = 10 */
(*y)++;         /* k++, k=k+1 = 10+1 = 11 */
```

Указателю-переменной можно присвоить значение другого указателя либо выражения типа указатель с использованием, при необходимости операции приведения типа (приведение обязательно, если один из указателей имеет тип "void *").

```
int i, *x;
char *y;
x=&i;           /* x -> поле объекта int */
y=(char *)x;   /* y -> поле объекта char */
y=(char *)&i;   /* y -> поле объекта char */
```

Рассмотрим фрагмент программы:

```
int a=5, *p, *p1, *p2;
p=&a; p2=p1=p;
++p1; p2+=2;
printf("a=%d, p=%d, p=%p, p1=%p, p2=%p.\n", a, p, p1, p2);
```

Результат выполнения:

```
a=5, *p=5, p=FFC8, p1=FFCC, p2=FFD0.
```

Конкретные значения адресов зависят от ряда причин: архитектура компьютера, тип и размер оперативной памяти и т.д.

Операции над указателями (адресная арифметика)

Записывая выражения и операторы, изменяющие значения указателей, необходимо помнить главное правило: *единицей изменения указателя является размер соответствующего ему типа*.

Например, `int *pti; char *ptc; float *ptf;`

Выполнение операторов `pti=pti+1;` или `pti++;` изменит значение указателя `pti` на 2, в результате чего он примет значение `FFC2`. В результате выполнения оператора `pti--;` значение указателя уменьшится на 2 и станет равным `FFBE`.

Аналогично для указателей других типов:

`ptc++;` увеличит значение указателя на 1;

`ptf++;` увеличит значение указателя на 4.

Разрешается сравнивать указатели и вычислять разность двух указателей. При сравнении могут проверяться отношения любого вида ("`>`", "`>=`", "`<`", "`<=`", "`==`", "`!=`"). Наиболее важными видами проверок являются отношения равенства или неравенства.

В заголовочном файле `stdio.h` определена константа – нулевой указатель с именем `NULL`. Ее значение можно присваивать указателю. Например: `ptf=NULL;`

Нулевой указатель обозначает отсутствие конкретного адреса ссылки.

Пример. Ввести два целых числа. Вычислить сумму, разность и деление этих чисел.

```
#include<stdio.h>
#include<stdio.h>
void main()
{ int a, b, *p1, *p2, *p3;
  float *p4;
  clrscr();
  puts("Введите 2 целых числа");
  scanf ("%d%d", a, b);
  *p1=a+b;
  *p2=a*b;
  *p3=a-b;
  *p4=(float)a/b;
  printf ("%d + %d=%d\n", a, b, *p1);
```

```
printf ("%d * %d=%d\n", a, b, *p2);
printf ("%d - %d=%d\n", a, b, *p3);
printf ("%d / %d=%f\n", a, b, *p4);
getch();
}
```

Порядок выполнения работы

1. Изучить теоретические сведения.
2. Ответить на контрольные вопросы.
3. Выполнить задание.

Контрольные вопросы

1. Что такое указатель? Для чего он применяется?
2. Как объявляется переменная типа указатель?
3. Почему при объявлении указателей необходимо указывать тип адресуемой с его помощью переменной?
4. Приведите пример инициализации указателя адресом переменной.
5. Каково назначение операции &?

Задания для выполнения

Применить технику указателей для решения следующих задач:

1. Ввести с клавиатуры три числа: X, Y, Z. Найти $\max(x + y + z, x*y*z) + 3$ и $\min(x^2 + y^2, y^2 + z^2) - 4$.
2. Ввести с клавиатуры четыре числа: A, B, C, D. Вычислить:
 - A + B + C + D, если A, B, C, D не равны между собой;
 - (A + B) * (C + D), если A, B и C, D попарно равны;
 - 4*A, если A, B, C, D между собой равны;
 - (A - B) * (C - D), в других случаях.
3. Ввести с клавиатуры три числа: A, B, C. Найти произведение двух наибольших чисел, воспользовавшись формулой $P = A * B * C / \min(A, B, C)$, где $\min(A, B, C)$ - наименьшее число из A, B, C.

4. Ввести с клавиатуры три числа и вывести следующее:
 0 - если числа не представляют стороны треугольника;
 1 - если числа представляют стороны треугольника, который не является ни равносторонним, ни равнобедренным;
 2 - если числа представляют стороны треугольника, который является равнобедренным;
 3 - если числа представляют стороны треугольника, который является равносторонним.

5. С клавиатуры введены три действительных числа. Определить, сколько среди них отрицательных.

6. На плоскости расположена окружность радиуса R с центром в начале координат. Ввести координаты точки и определить, лежит ли она на окружности.

7. Даны площади круга R и квадрата S . Определить поместится ли квадрат в круге.

8. На плоскости расположена окружность радиуса R с центром в точке (a, b) . Ввести заданные координаты точки и определить ее расположение относительно окружности.

9. На плоскости расположено кольцо с центром в начале координат, внутренним радиусом r и внешним радиусом R . Ввести координаты точки и определить, попадет ли она в кольцо.

10. Даны координаты вершин треугольника. Определить, является ли он

а) прямоугольным;

б) равносторонним.

11. Даны координаты вершин двух треугольников. Определить, какой из них имеет наибольшую площадь.

12. Число X введено с клавиатуры. Вывести на экран в порядке возрастания числа

$$\frac{e^x + e^{-x}}{2}, 1 + |x|, (1 + x^2)^x.$$

13. Определить, принадлежит ли точка D треугольнику ABC , который задан координатами своих вершин. Координаты

вершин треугольника и координаты точки D вводятся с клавиатуры.

14. Даны площади круга R и квадрата S . Определить, поместится ли круг в квадрате.

15. На плоскости расположена окружность радиуса R с центром в начале координат. Ввести координаты точки и определить ее расположение относительно окружности.