

ГЛАВА 8

СТРУКТУРЫ

Целью данной лабораторной работы является приобретение навыков в определении и использовании структур. Для более уверенного овладения материалом рекомендуется решить 2 - 3 задачи.

ПРИМЕРЫ РЕШЕНИЯ ЗАДАЧ

Задача А

Определить структуру, описывающую понятие *комплексное число*. Написать и протестировать функции для ввода, вывода комплексного числа, а также для сложения двух комплексных чисел, заданных в алгебраической форме.

```
#include <stdio.h>

typedef struct
{
    int real;
    int imag;
} Complex;

void main()
{
    Complex c1, c2, c3, read();
    void add(Complex, Complex, Complex*), print(Complex);

    c1=read(); c2=read();
    add(c1, c2, &c3);
    printf("При сложении "); print (c1); printf(" и "); print (c2);
    printf("\n получилось "); print(c3);
}

Complex read ()
{
    Complex c;
    puts("Введите действительную и мнимую части числа:");
    scanf("%d %d", &(c.real), &(c.imag));
    return c;
}

void print (Complex c)
{ printf("%d+i*(%d)", c.real, c.imag); }

void add (Complex c1, Complex c2, Complex* c3)
{
```

```
c3->real=c1.real+c2.real;
c3->imag=c1.imag+c2.imag;
}
```

Задача В

Определить структуру, описывающую понятие *рациональное число*. Написать и протестировать функции для сокращения, печати рационального числа, а также для деления двух рациональных чисел.

```
#include<stdio.h>
#include<math.h>

typedef struct
    {
        int chis;
        int znam;
    } Racion;

void put(Racion A)
{
    if(A.chis*A.znam<0) printf("-");
    A.chis=abs(A.chis);
    A.znam=abs(A.znam);
    printf("%d/%d", A.chis, A.znam);
}

Racion SOKR (Racion A)
{
    int i, min;

    if(abs(A.chis) > abs(A.znam)) min=abs(A.znam);
    else min=abs(A.chis);
    for(i=min; i>1; i--)
        if(A.chis%i == 0 && A.znam%i == 0) break;
    A.chis/=i;
    A.znam/=i;
    return A;
}

void DIV (Racion A, Racion B, Racion *C)
{
    A=SOKR(A);
    B=SOKR(B);
    C->chis=A.chis*B.znam;
    C->znam=A.znam*B.chis;
    *C=SOKR(*C);
}
```

```
void main()
{ Racion A, B, C;

  puts("\nВведите числитель и знаменатель первой, а затем - второй дроби:");
  scanf("%d%d%d%d", &A.chis, &A.znam, &B.chis, &B.znam);
  DIV(A, B, &C);
  printf("При делении "); put(A); printf(" на "); put(B);
  printf("\n получилось "); put(C);
} □
```

Задача С

По введенным году, месяцу и числу определить порядковый номер дня в году.

```
typedef struct
{
  int year;
  int month;
  int day;
} DATE;

int tab_days[2][13]={
                {0,31,28,31,30,31,30,31,31,30,31,30,31}
                {0,31,29,31,30,31,30,31,31,30,31,30,31}
                };

int day_of_year (DATE d)
{
  int i, k;
  k=d.year%4==0 && d.year%100!=0 || d.year%400==0; // високосный ли год
  if(d.month<1 || d.month>12) return -1;
  if(d.day<1 || d.day>tab_days[k][d.month]) return -1;
  for(i=1; i<d.month; i++)
    d.day+=tab_days[k][i];
  return d.day;
}

#include<stdio.h>
void main()
{
  DATE d;
  int N;

  puts("Введите число, номер месяца и год.");
  scanf("%d%d%d", &d.day, &d.month, &d.year);
  N=day_of_year(d);
```

```
if(N<0) puts("Неверно введена      дата!");  
else printf("В %d году этот день имеет номер %d\n", d.year, N);  
}
```

ЗАДАНИЕ НА

ПРОГРАММИРОВАНИЕ

Задача 1

Ввести перечислимые типы *масть*, *достоинство*. С их помощью описать как структуру переменную *карта*. Составить и протестировать функцию БЬЁТ (K1, K2, KM), которая проверяет, бьёт ли карта K1 карту K2, с учетом того, что масть KM является козырной.

Задача 2

Описать как структуру переменную *время* (с полями часы, минуты, секунды). Составить и протестировать функции:

а) СЛЕД_СЕК (t, t1, d),

которая присваивает параметру t1 время на d секунд большее, чем время t (может происходить смена суток);

б) ИНТЕРВАЛ (t1, t2, d),

которая вычисляет время d, прошедшее от времени t1 до времени t2.

Задача 3

Ввести перечислимые типы *вертикаль*, *горизонталь* для обозначения клеток шахматной доски. Составить и протестировать функции:

а) ХОД_ФЕРЗЯ (K1, K2),

которая проверяет, может ли ферзь за один ход перейти с поля K1 на поле K2;

б) ХОД_КОНЯ (K1, K2),

которая вычисляет, за сколько ходов конь может перейти с поля K1 на поле K2.

Задача 4

Ввести структуру (с полями *числитель* и *знаменатель*) для описания понятия *рациональное число*. Составить и протестировать функции:

а) РАВНО (A, B),

которая проверяет, равны ли друг другу рациональные числа A, B;

б) МАКС (X, N),

которая возвращает наибольшее из массива X[N] рациональных чисел;

в) СЛОЖ (A, B, C),

которая записывает в C результат сложения рациональных чисел A и B;

г) МИН (A, B)

которая возвращает наименьшее из двух рациональных чисел A и B;

д) УМН (A, B, C),

которая записывает в C результат перемножения рациональных чисел A и B.

Задача 5

Ввести структуру (с полями *число*, *месяц*, *год*) для описания понятия *дата*. Составить и протестировать функцию, которая

- а) вычисляет интервал (в днях), прошедший между двумя датами;
- б) по порядковому номеру дня в году определяет число и месяц года, соответствующие этому дню;
- в) по введенной дате распечатывает дату на N дней вперед.

Задача 6

Определить структуры, описывающие *шар* и *точку* в трехмерном пространстве. Составить и протестировать функцию, которая проверяет, находится ли точка внутри заданного шара.

Задача 7

Ввести структуру для описания *комплексного числа*. Составить и протестировать функции для:

- а) преобразования комплексного числа из алгебраической формы в показательную;
- б) преобразования комплексного числа из показательной формы в алгебраическую;
- в) получения сопряженного комплексного числа;
- г) возведения комплексного числа в целую положительную степень;
- д) умножения комплексных чисел в алгебраической форме;
- е) умножения комплексных чисел в показательной форме;
- ж) деления комплексных чисел в показательной форме;
- з) деления комплексных чисел в алгебраической форме.

Задача 8

Ввести структуру для описания понятия *алгебраический полином*. Составить и протестировать функции для:

- а) ввода полинома;
- б) вывода полинома;
- в) нормализации полинома;
- г) сложения полиномов;
- д) вычитания полиномов;
- е) умножения полиномов;
- ж) деления полиномов;
- з) дифференцирования полинома;
- и) интегрирования полинома.

Задача 9

Ввести структуру для регистрации автомашин. Она должна иметь следующие поля:

- дата регистрации (структура с полями - день, месяц, год);
- марка машины;
- год выпуска;

- цвет;
- номер.

Написать и протестировать функции:

- регистрации новой машины;
- удаление машины из регистрационного списка;
- поиск машины по любой из комбинаций признаков.

Задача 10

Массив структур содержит информацию о студентах группы: в первом поле стоит фамилия, во втором - возраст, в третьем - рост, в четвертом - средний балл в сессию и т.д. (i -ый элемент массива описывает i -го студента).

Студент называется *среднестатистическим* по k -му параметру, если на нем достигается *минимум модуля разности* среднего арифметического чисел k -го столбца и значения k -го параметра этого студента. Аналогично определяется *уникальный* по k -му параметру студент (на нем достигается *максимум*).

Студент называется *самым средним*, если он является среднестатистическим по самому большому количеству параметров. Аналогично определяется *самый уникальный* студент.

Выяснить, кто в группе является:

- а) самым средним,
- б) самым уникальным,
- в) самым средним среди самых уникальных,
- г) самым уникальным среди самых средних.

Задача 11

В доме N этажей и три лифта. Каждый лифт может быть свободным или занятым. Человек стоит на одном из этажей и собирается вызвать либо ближайший свободный лифт, либо ближайший занятый, направляющийся в сторону этажа, где находится человек.

Распечатать начальную конфигурацию (расстановку, занятость и направление движения лифтов, местоположение человека), а также номер лифта, который будет вызван.

(Использовать функции ВВОД, ВЫВОД, ВЫБОР_ЛИФТА).

Задача 12

Пусть ЭВМ не умеет работать с вещественными числами, а имеет только операции и функции для работы с символами, строками и целыми числами.

Реализовать функции для:

- а) ввода;
- б) вывода;
- в) сложения;
- г) вычитания;

д) умножения вещественных чисел. (Числа вводятся как строки, разделяются на целую и дробную части, и над ними, как над целыми числами, с учетом межразрядных переносов, выполняются операции).

Задача 13

Определить структуру, описывающую равнобедренный прямоугольный треугольник с катетами, параллельными осям координат, и нижним левым прямым углом.

Написать и протестировать функцию, возвращающую указатель на новый треугольник - область пересечения двух заданных. Если пересечения нет - возвращается NULL.

Задача 14

Определить структуры, описывающие *точку* в полярной и декартовой системах координат. Составить и протестировать функции для:

- а) получения декартовых координат точки, если заданы ее полярные координаты;
- б) вычисления расстояния между двумя точками, заданными в декартовой системе координат;
- в) получения полярных координат точки, если заданы ее декартовы координаты;
- г) вычисления расстояния между двумя точками, заданными в полярной системе координат.

Задача 15

Определить структуру - важнейшие исторические даты. Её поля - год,- событие.

Написать и протестировать функции:

- сортирующую структуры по любому из полей;
- подсчитывающую средний интервал между датами;
- определяющую наиболее часто встречающуюся первую букву в названии события.

Задача 16

Определить структуру, описывающую прямоугольник со сторонами, параллельными осям координат (прямоугольник задаётся двумя точками - левой нижней и правой верхней).

Написать и протестировать функцию, возвращающую указатель на новый прямоугольник - область пересечения двух прямоугольников. Если пересечения нет - возвращается NULL.