# Лабораторная работа № 7. Условные операторы

## Задание 1

1. С клавиатуры вводятся три числа. Определите большее из трех чисел.
2. Следует сравнивать каждое из чисел с двумя другими, т. к. заранее не известно какое из чисел будет наибольшим.

|  |  |  |
| --- | --- | --- |
| Идентификатор | Значение | Тип |
| a, b, c | Числа | int |

#include <stdlib.h>  
#include <stdio.h>  
  
int main() {  
 int a, b, c;  
 printf("Введите три числа: ");  
 scanf("%d %d %d", &a, &b, &c);  
 if ((a >= b) && (b >= c)) {  
 printf("Наибольшее число: %d.", a);  
 } else if ((b >= a) && (b >= c)) {  
 printf("Наибольшее число: %d.", b);  
 } else if ((c >= a) && (c >= b)) {  
 printf("Наибольшее число: %d.", c);  
 }  
 return 0;  
}

1. 

## Задание 2

1. Ввести . Если , вычислить площадь прямоугольника; при , найти площадь квадрата; в противном случае подсчитать площадь трапеции.
2. Площадь прямоугольника, считается также, как и площадь квадрата — . Площадь трапеции . Для того чтобы определить какую площадь необходимо вычислить нужно проверять значения переменных .

|  |  |  |
| --- | --- | --- |
| Идентификатор | Значение | Тип |
| a, b, h | Стороны прямоугольника, квадрата или трапеции | float |
| s | Площадь | float |

#include <stdlib.h>  
#include <stdio.h>  
  
int main() {  
 float a, b, h, s;  
 printf("Введите стороны фигуры — a, b, h: ");  
 scanf("%f %f %f", &a, &b, &h);  
 if (h == 0) {  
 s = a \* b;  
 if (a == b)  
 printf("Площадь квадрата: %4.2f.", s);  
 else  
 printf("Площадь прямоугольника: %4.2f.", s);  
 } else {  
 s = (a + b) / 2 \* h;  
 printf("Площадь трапеции: %4.2f.", s);  
 }  
 return 0;  
}

1. 

## Задание 3

1. Ввести с клавиатуры цифру. Определить, какой системе счисления она может принадлежать.
2. В системе счисления -го порядка могут использоваться все цифры до включая. принадлежит любой системе счисления.

|  |  |  |
| --- | --- | --- |
| Идентификатор | Значение | Тип |
| x | Число, которое вводится с клавиатуры | int |

#include <stdlib.h>  
#include <stdio.h>  
  
int main() {  
 int x;  
 printf("Введите цифру: ");  
 scanf("%d", &x);  
 if (x == 0)  
 printf("Цифра %d принадлежит любой системе счисления.", x);  
 else  
 printf("Цифра %d принадлежит %d-ой системе счисления и выше.", x, x + 1);  
 return 0;  
}

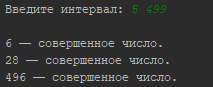
1. 

## Задание 4

1. Программа позволяет в заданном интервале найти все совершенные числа. Натуральное число называется совершенным, если оно равно сумме всех своих делителей, не считая его самого.
2. Совершенное число — натуральное число, равное сумме всех своих собственных делителей (то есть всех положительных делителей, отличных от самого́ числа). Примеры совершенных чисел: 6, 28, 496, 8128, 33 550 336, 8 589 869 056, 137 438 691 328.

|  |  |  |
| --- | --- | --- |
| Идентификатор | Значение | Тип |
| a, b | Интервал, в котором ищутся совершенные числа | int |
| t | Сумма делителей | int |
| i, j | Счётчики циклов | int |

#include <stdlib.h>  
#include <stdio.h>  
#include <math.h>  
  
int main() {  
 int a, b; /\* a, b — границы интервалов \*/  
 printf("Введите интервал: ");  
 scanf("%d %d", &a, &b);  
 int t = 0; /\* В t будет храниться сумма делителей \*/  
 for (int i = a; i <= b; i++) {  
 for (int j = 1; j < i; j++) {  
 if (i % j == 0) {  
 t += j;  
 }  
 }  
 if (t == i) {  
 printf("\n%d — совершенное число.", i);  
 }  
 t = 0;  
 }  
 return 0;  
}

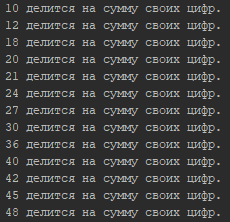
1. 

## Задание 5

1. Определить двузначные целые числа, которые делятся на сумму своих цифр.
2. Сумма цифр вычисляется с помощью целочисленного деления числа на 10 и суммирования остатка от этого деления.

|  |  |  |
| --- | --- | --- |
| Идентификатор | Значение | Тип |
| i | Счётчик и параметр циклов | int |
| sum | Сумма цифр | int |
| t | Временная переменная для того, чтобы сохранить значение двузначного числа | int |

#include <stdlib.h>  
#include <stdio.h>  
#include <math.h>  
  
int main() {  
 int i, t, sum;  
 for (i = 10; i < 50; i++) {  
 sum = 0;  
 t = i;  
 while (i != 0) {  
 sum += i % 10;  
 i = i / 10;  
 }  
 i = t;  
 if (i % sum == 0)  
 printf("%d делится на сумму своих цифр.\n", i);  
 }  
 return 0;  
}

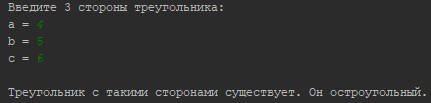
1. 

## Задание 6

1. Составьте программу, которая по трём введённым числам определяет, могут ли быть эти числа длинами сторон треугольника. если да, то какой получится треугольник с данными сторонами (прямоугольный, остроугольный, тупоугольный).
2. Если каждая сторона треугольника меньше суммы двух других сторон, то такой треугольник существует. Для того, чтобы по сторонам определить тупоугольный ли треугольник, остроугольный или прямоугольный можно найти косинус наибольшего угла по теореме косинусов (чем больше угол, тем меньше будет косинус, поэтому среди всех косинусов надо искать наименьший), и если он равен 0, то треугольник прямоугольный, если меньше нуля — тупоугольный, если больше — остроугольный.

|  |  |  |
| --- | --- | --- |
| Идентификатор | Значение | Тип |
| a, b, c | Стороны треугольника | float |
| smallest(a, b, c) | Определяет наименьшее число | float |
| cosinus(a, b, c) | Находит косинус наибольшего угла по теореме косинусов | float |

#include <stdlib.h>  
#include <stdio.h>  
#include <math.h>  
  
/\* Функция находит наименьшее число \*/  
float smallest(float a, float b, float c) {  
 if ((a <= b) && (b <= c)) {  
 return a;  
 } else if ((b <= a) && (b <= c)) {  
 return b;  
 } else if ((c <= a) && (c <= b)) {  
 return c;  
 }  
}  
/\* Функция находит косинус наибольшего угла \*/  
float cosinus(float a, float b, float c) {  
 float cos1, cos2, cos3;  
 cos1 = (a \* a + b \* b - c \* c) / (2 \* a \* b);  
 cos2 = (c \* c + a \* a - b \* b) / (2 \* c \* a);  
 cos3 = (c \* c + b \* b - a \* a) / (2 \* c \* b);  
 return smallest(cos1, cos2, cos3);  
}  
  
int main() {  
 float a, b, c;  
 printf("Введите 3 стороны треугольника:\n");  
 printf("a = ");  
 scanf("%f", &a);  
 printf("b = ");  
 scanf("%f", &b);  
 printf("c = ");  
 scanf("%f", &c);  
 /\* Если косинус наибольшего угла = 0 — треугольник прямоугольный,  
 если косинус > 0 — остроугольный, если < 0 — тупоугольный \*/  
 if ((a < b + c) && (b < a + c) && (c < a + b)) {  
 printf("\nТреугольник с такими сторонами существует.");  
 if (cosinus(a, b, c) > 0) {  
 printf(" Он остроугольный.");  
 }  
 if (cosinus(a, b, c) == 0) {  
 printf(" Он прямоугольный.");  
 }  
 if (cosinus(a, b, c) < 0) {  
 printf(" Он тупоугольный.");  
 }  
 } else {  
 printf("\nТакого треугольника не существует.");  
 }  
 return 0;  
}

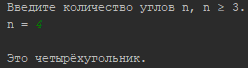
1. 

## Задание 7

1. Напечатать в зависимости от числа углов название фигуры (треугольник, четырёхугольник, пятиугольник, шестиугольник, многоугольник). Вариант реализации задачи написать с применением оператора выбора.
2. Количество углов у фигуры должно быть не менее 3-х.

|  |  |  |
| --- | --- | --- |
| Идентификатор | Значение | Тип |
| n | Количество углов | int |

#include <stdlib.h>  
#include <stdio.h>  
#include <math.h>  
  
int main() {  
 int n;  
 printf("Введите количество углов n, n ≥ 3.\nn = ");  
 scanf("%d", &n);  
 switch (n) {  
 case 3:  
 printf("\nЭто треугольник.");  
 break;  
 case 4:  
 printf("\nЭто четырёхугольник.");  
 break;  
 case 5:  
 printf("\nЭто пятиугольник.");  
 break;  
 case 6:  
 printf("\nЭто шестиугольник.");  
 break;  
 default:  
 printf("\nЭто многоугольник.");  
 }  
 return 0;  
}

1. 

## Задание 8

1. В зависимости от номера () типа фигуры, организовать ввод необходимых данных и вычислить при — площадь круга (), — объем шара (), — объем цилиндра (), — площадь поверхности сферы ().
2. , , , .

|  |  |  |
| --- | --- | --- |
| Идентификатор | Значение | Тип |
| n | Номер фигуры | int |
| r | Радиус | float |
| h | Высота цилиндра | float |
| S | Результат | float |

#include <stdlib.h>  
#include <stdio.h>  
#include <math.h>  
  
int main() {  
 int n;  
 float r, s, h;  
 printf("Что будем искать? (введите только число)\n1. Площадь круга\n2. Объём шара\n"  
 "3. Объём цилиндра\n4. Площадь поверхности сферы\n");  
 scanf("%d", &n);  
 if (n > 0 && n < 5) {  
 printf("Введите радиус.\nr = ", r);  
 scanf("%f", &r);  
 }  
 switch (n) {  
 case 1:  
 s = M\_PI \* r \* r;  
 break;  
 case 2:  
 s = 4 / 3 \* M\_PI \* r \* r \* r;  
 break;  
 case 3:  
 printf("Введите высоту цилиндра.\nh = ");  
 scanf("%f", &h);  
 s = M\_PI \* r \* r \* h;  
 break;  
 case 4:  
 s = 4 \* M\_PI \* r \* r;  
 break;  
 default:  
 printf("Ошибка. Нужно было ввести число от 1 до 4.\n");  
 }  
 printf("s = %f", s);  
 return 0;  
}

1. 