**Лабораторная работа №6**

“Динамические массивы”

Задание 7.1

**Постановка задачи:** С клавиатуры вводятся три числа. Определите большее из трех чисел.

**Математическая модель:** Заранее не известно какое из чисел будет наибольшим, поэтому следует сравнивать каждое из чисел с двумя другими.

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Идентификатор** | **Семантика** | **Тип** |
| a, b, c | Три числа, которые мы будем сравнивать | int |

**Код программы:**

include <stdio.h>

#include <stdlib.h>

int main(int argc, char const \*argv[]) {

int a, b, c;

printf("Enter three numbers:");

scanf("%d %d %d", &a, &b, &c);

if ((a >= b) && (b >= c)) {

printf("The biggest one is:%d\n", a);

} else if ((b >= a) && (b >= c)) {

printf("The biggest one is:%d\n", b);

} else if ((c >= a) && (c >= b)) {

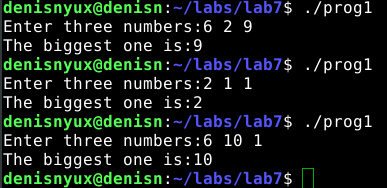
printf("The biggest one is:%d\n", c);

}

return 0;

}

**Результат выполненной работы:**

****

**Задание 7.2**

**Постановка задачи:** Ввести a, b, h. Если h=0, вычислить площадь прямоугольника; при a = b, найти площадь квадрата; в противном случае подсчитать площадь трапеции.

**Математическая модель:** Площадь прямоугольника, считается точно также как и площадь квадрата: s = a \* b. Площадь трапеции s = h \* (a + b) / 2. Для того чтобы определить какую площадь необходимо вычислить нужно проверять значения переменных a, b, h.

**Список идентификаторов:**

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

**Код программы:**

#include <stdio.h>

#include <stdlib.h>

int main(int argc, char const \*argv[]) {

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;

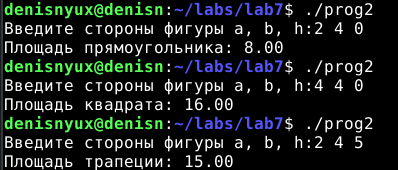
printf("Площадь трапеции: %4.2f", s);

}

return 0;

}

**Результат выполненной работы:**

****

**Задание 7.3**

**Постановка задачи:** Ввести с клавиатуры цифру. Определить, какой системе счисления она может принадлежать.

**Математическая модель:** В системе счисления n порядка могут использоваться все цифры до n-1 включая. 0 при надлежит любой системе счисления.

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Идентификатор** | **Семантика** | **Тип** |
| x | Число, которое вводится | int |

**Код программы:**

#include <stdio.h>

#include <stdlib.h>

int main(int argc, char const \*argv[]) {

int x;

printf("Введите цифру: ");

scanf("%d", &x);

if (x == 0)

printf("%d принадлежит любой системе счисления\n", x);

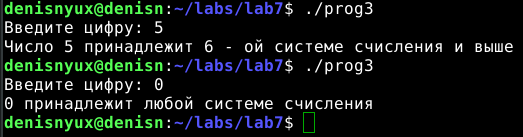
else

printf("Число %d принадлежит %d - ой системе счисления и выше\n", x, x + 1);

return 0;

}

**Результат выполненной работы:**

****

**Задание 7.4**

**Постановка задачи:** Программа позволяет в заданном интервале найти все совершенные числа. Натуральное число называется совершенным, если оно равно сумме всех своих делителей, не считая его самого.

**Математическая модель:** Совершенное число равно сумме всех своих делителей, не считая его самого. Несколько совершенных чисел: 6, 28, 496, 8128, 33 550 336, 8 589 869 056, 137 438 691 328 и т.д.

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Идентификатор** | **Семантика** | **Тип** |
| a, b | Интервал, в котором ищутся совершенные числа | int |

**Код программы:**

#include <math.h>

#include <stdio.h>

#include <stdlib.h>

int main(int argc, char const \*argv[]) {

int a, b; /\*a,b - границы интервалв\*/

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("%d - совершенное\n", i);

}

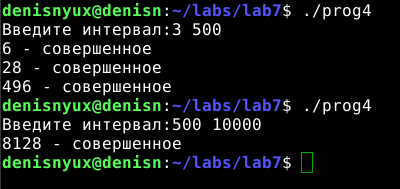
t = 0;

}

return 0;

}

**Результат выполненной работы:**

****

**Задание 7.5**

**Постановка задачи:** Определить двузначные целые числа, которые делятся на сумму своих цифр.

**Математическая модель:** Сумма цифр вычисляется с помощью целочисленного деления числа на 10 и суммирования остатка от этого деления.

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Идентификатор** | **Семантика** | **Тип** |
| i | Параметр циклов | int |
| sum | Сумма цифр | int |
| t | Временная переменная для того чтобы сохранить значение двузначного числа | int |

**Код программы:**

#include <math.h>

#include <stdio.h>

#include <stdlib.h>

int main(int argc, char const \*argv[]) {

int i, t, sum;

for (i = 10; i < 100; i++) {

sum = 0;

t = i;

while (i != 0) {

sum += i % 10;

i = i / 10;

}

i = t;

if (i % sum == 0)

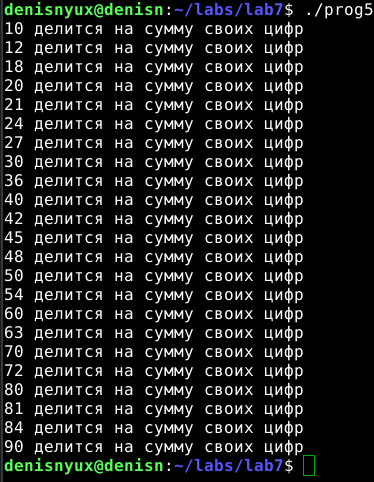
printf("%d делится на сумму своих цифр\n", i);

}

return 0;

}

**Результат выполненной работы:**

****

**Задание 7.6**

**Постановка задачи:** Составьте программу, которая по трем введенным числам определяет, могут ли быть эти числа длинами сторон треугольника. если да, то какой получится треугольник с данными сторонами (прямоугольный, остроугольный, тупоугольный).

**Математическая модель:** 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.

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Идентификатор** | **Семантика** | **Тип** |
| lowest(a,b,c) | Определяет наименьшее число | float |
| cosinus(a,b,c) | Находит косинус наибольшего угла по теореме косинусов | float |
| a, b, c | Стороны | float |

**Код программы:**

#include <math.h>

#include <stdio.h>

#include <stdlib.h>

/\*Функция находит наименьшее число среди 3\*/

float lowest(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 lowest(cos1, cos2, cos3);

}

float main(int argc, char const \*argv[]) {

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);

/\*

printf("%f\n", lowest(a, b, c));

printf("%f\n", cosinus(a, b, c));

\*/

/\*Если косинус наибольшего угла равен 0 - треугольник прямоугольный,

если косинус больше 0 - остроугольный, если меньше 0 - тупоугольный\*/

if ((a < b + c) && (b < a + c) && (c < a + b)) {

printf("Треугольник с такими сторонами существует.");

if (cosinus(a, b, c) > 0) {

printf(" Он остроугольный\n");

}

if (cosinus(a, b, c) == 0) {

printf(" Он прямоугольный\n");

}

if (cosinus(a, b, c) < 0) {

printf(" Он тупоугольный\n");

}

} else {

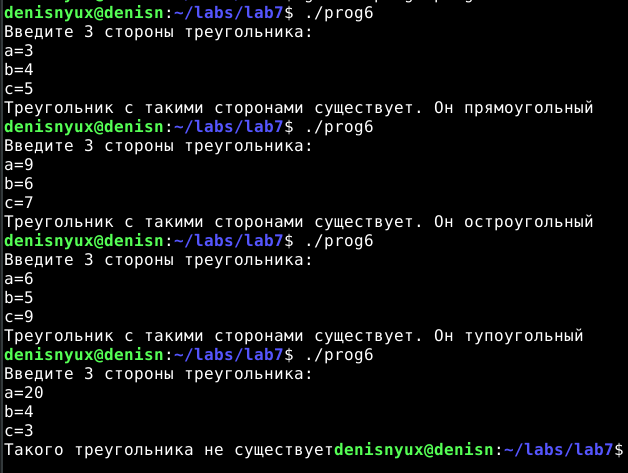
printf("Такого треугольника не существует");

}

return 0;

}

**Результат выполненной работы:**

****

**Задание 7.7**

**Постановка задачи:** Напечатать в зависимости от числа углов название фигуры (треугольник,четырехугольник, пятиугольник, шестиугольник, многоугольник). Вариант реализации задачи написать с применением оператора выбора.

**Математическая модель:** Количество углов у фигуры не может быть меньше 3.

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Идентификатор** | **Семантика** | **Тип** |
| n | Количество углов | int |

**Код программы:**

#include <math.h>

#include <stdio.h>

#include <stdlib.h>

int main(int argc, char const \*argv[]) {

int n;

printf("Введите количество углов n, n>2\nn=");

scanf("%d", &n);

switch (n) {

case 3:

printf("Треугольник\n");

break;

case 4:

printf("Четырехугольник\n");

break;

case 5:

printf("Пятиугольник\n");

break;

case 6:

printf("Шестиугольник\n");

break;

default:

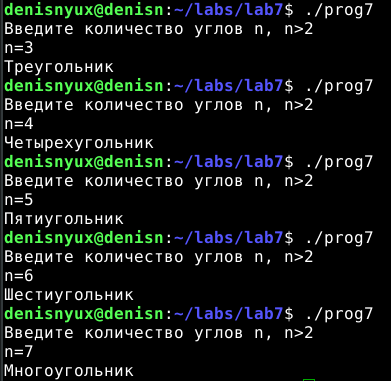
printf("Многоугольник\n");

}

return 0;

}

**Результат выполненной работы:**

****

**Задание 7.8**

**Постановка задачи:** В зависимости от номера (N) типа фигуры, организовать ввод необходимых данных и вычислить при N = 1 - площадь круга, N = 2- объем шара (4/3πR3), N=3 - объем цилиндра, N = 4-площадь поверхности сферы 4πr2.

**Математическая модель:** Площадь круга s = π \* r^2, объем шара s = 4/3\*π\*R^3, объем цилиндра s = π \* r \* r \* h, площадь поверхности сферы s = 4 \* π \* r^2.

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Идентификатор** | **Семантика** | **Тип** |
| n | Номер фигуры | int |
| r | Радиус | float |
| h | Высота цилиндра | float |

**Код программы:**

#include <math.h>

#include <stdio.h>

#include <stdlib.h>

int main(int argc, char const \*argv[]) {

int n;

float r, s, h;

printf("Что будем искать?\n1)Площадь круга\n2)Объем шара\n3)Объем "

"цилиндра\n4)Площадь поверхности сферы\n");

scanf("%d", &n);

if (n < 5 && n > 0) {

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 циллиндра\nh=");

scanf("%f", &h);

s = M\_PI \* r \* r \* h;

break;

case 4:

s = 4 \* M\_PI \* r \* r;

break;

default:

printf("Ошибка\n");

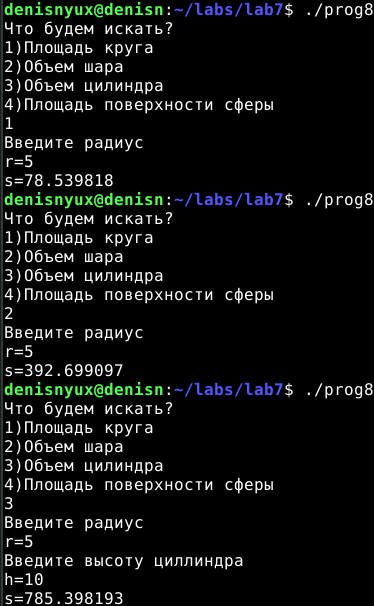
}

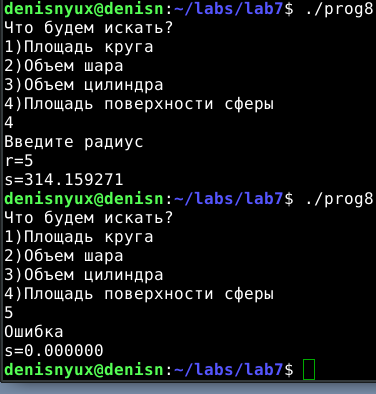
printf("s=%f\n", s);

return 0;

}

**Результат выполненной работы:**

****

****