

**Нахождение элементов
треугольника.**

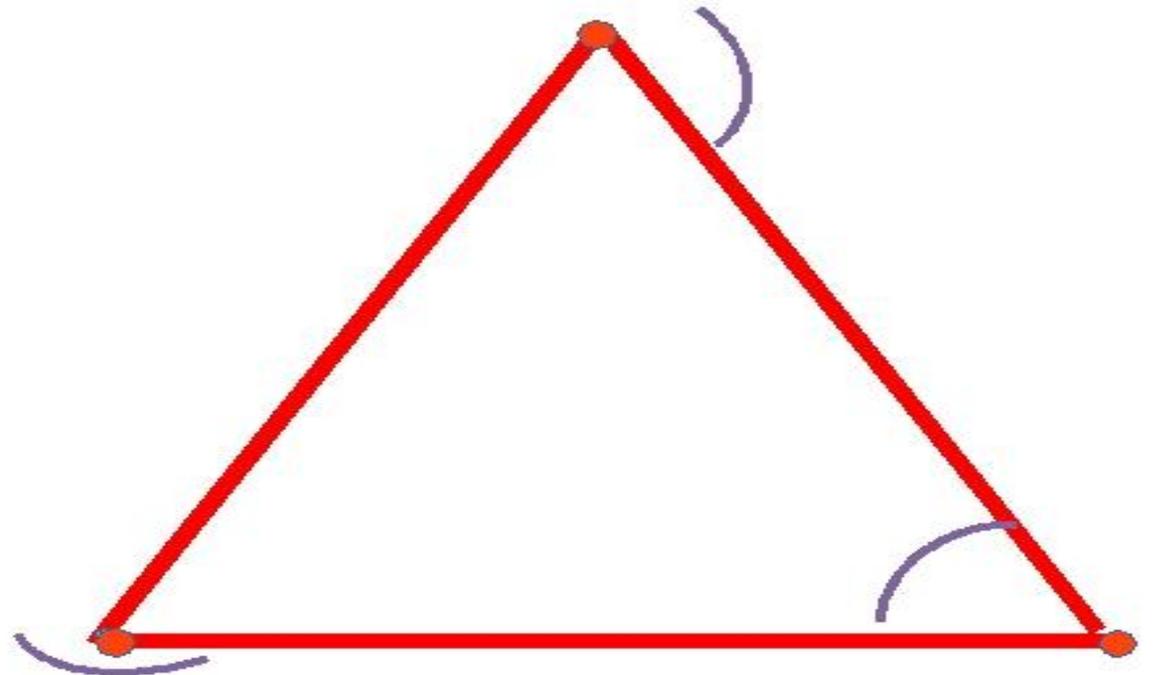
Идея разработки.

- Идея разработки: разработка обучающей программы по нахождению элементов треугольника.
- Входными данными являются координаты точек-углов треугольника.
- На выходе: длины сторон, высота, площадь, периметр, средняя линия, медиана, радиус вписанной окружности, радиус описанной окружности. Все они обязательно должны быть с расчетными формулами.

определение треугольника.

Что такое треугольник?

- Геометрическая фигура, у которой:
- **три вершины;**
- **три угла,**
- **три стороны**



Формулы элементов

Треугольника

- 1) Длина стороны: $AB = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} = \sqrt{a^2 + b^2}$.
- 2) Высота: $h = (BC * AC) / 2R$.
- 3) Площадь: $S = (AB * BC * AC) / 4 * R$.
- 4) Периметр: $P = AB + BC + AC$.
- 5) Средняя линия: $KM = AC / 2$.
- 6) Медиана: $m = (\sqrt{2 * AB * AB + 2 * BC * BC - AC * AC}) / 2$.
- 7) Радиус вписанной окружности: $r = (2 * S) / P$.
- 8) Радиус описанной окружности: $R = (AB * BC * AC) / (4 * S)$.

Список требований.

- 1) Группы пользователей:
 - -Учащиеся
 - -Учителя
- 2)Функциональные требования:
 - Это функции элементов треугольника
- 3)Требования к данным:
 - -В состав входной информации должны приниматься значения координат точек-углов, а в состав выходной информации должны приниматься значения элементов треугольника, вычисленных по данным формулам.
 - -Ввод/ вывод данных осуществляется только с клавиатуры.
 - -Все числа или цифры в вводе/выводе могут быть любыми, и целыми, и дробными.

Список требований

4) Требования к интерфейсу:

Пользователь должен взаимодействовать с программой через консоль.

5) Требования к окружению.

Данных требований к окружению не выделено.

Проектирование

- 1) Проектирование функций:
 - Функции проектируются за счет входных и выходных данных наших элементов треугольника.
- 2) Проектирование данных:
 - Тут мы работаем с проектированием входных и выходных данных.
- 3) Проектирование интерфейса:
 - Это описание того, что будет видеть пользователь при работе с программой.

Программная реализация

- В качестве примера программной реализации разберем 2 функции основного кода.
- `#include <stdio.h>`
- `#include <math.h>`
- `float length (float x1, float y1, float x2, float y2){`
 - `float a = x2 - x1;`
 - `float b = y2 - y1;`
 - `float len = sqrt(a * a + b * b);`
 - `return len;`
 - `}`

```
float perimeter(float AB, float BC, float AC){
```

```
    float P= AB+BC+AC;
```

```
    return P;
```

```
}
```

```
int main(){
```

```
    float Xa, Ya;
```

```
    float Xb, Yb;
```

```
    float Xc, Yc;
```

```
    float P;
```

```
    float AB, BC, AC;
```

```
        printf("Enter the coordinates of A: ");
```

```
    scanf("%f %f", &Xa, &Ya);
```

```
        printf("Enter the coordinates of B: ");
```

```
    scanf("%f %f", &Xb, &Yb);
```

```
        printf("Enter the coordinates of C: ");
```

```
    scanf("%f %f", &Xc, &Yc);
```

```
float lenAB = length(Xa, Ya, Xb, Yb);
```

```
float lenBC = length(Xc, Yc, Xb, Yb
```

```
float lenAC = length(Xa, Ya, Xc, Yc);
```

```
printf("lenAB=%f lenBC=%f lenAC=%f\n", lenAB, lenBC, lenAC);
```

- `printf("Perimeter=%f\n", perimeter(lenAB,lenBC,lenAC));`
- `return 0;`
- `}`