

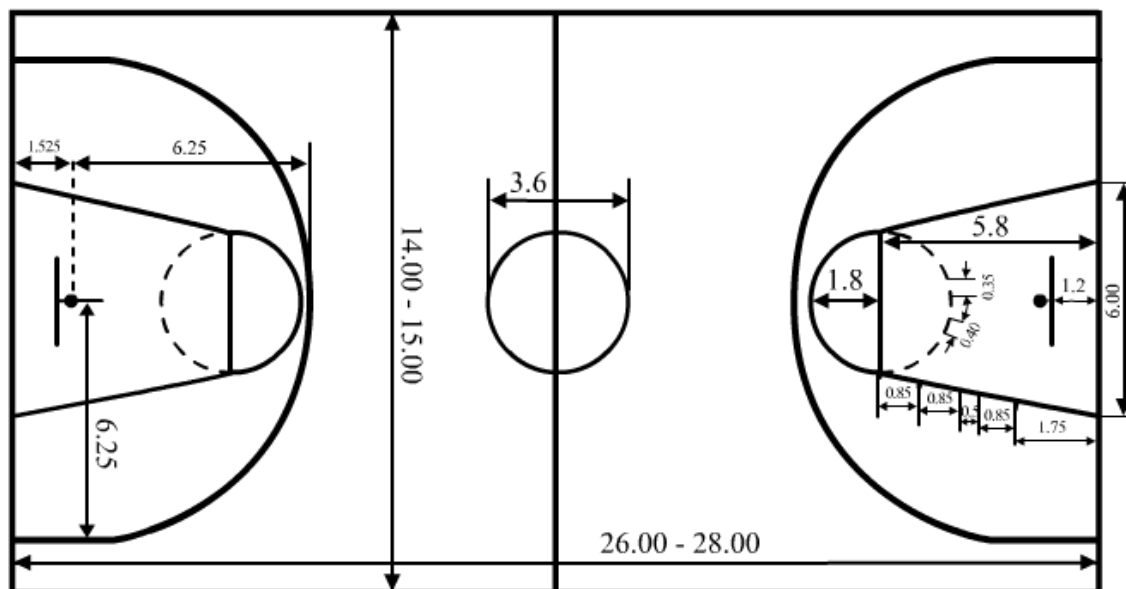
Задание 1

Построим модель броска мяча в корзину при игре в баскетбол. Необходимо определить по начальной скорости и углу и росту учащегося расстояние, на котором он должен находиться, чтобы брошенный им мяч попал в корзину.

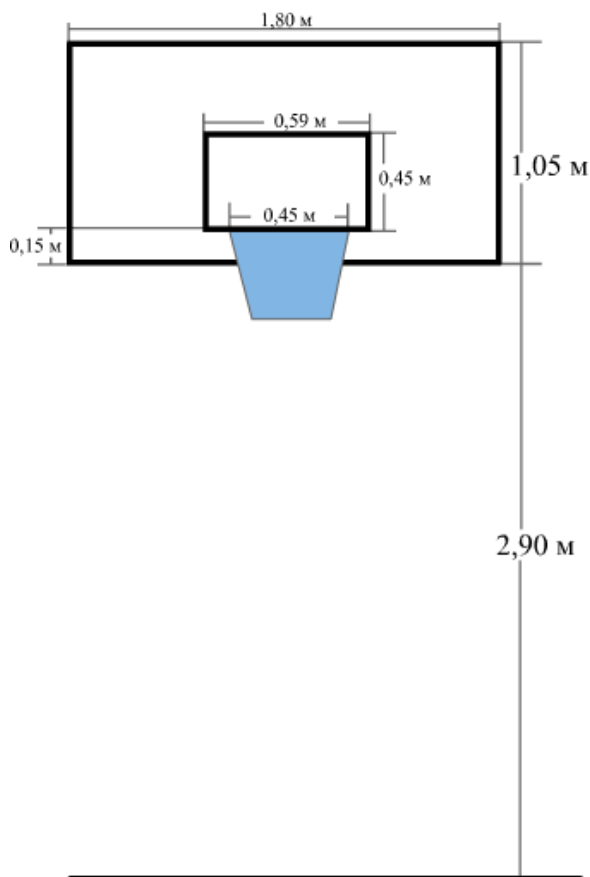
На уроке физкультуры ученики по очереди кидают мяч в кольцо, становясь как можно дальше от кольца. Так как все ученики разного роста, то и стоять они будут на разном расстоянии.



По правилам игры установлены следующие размеры площадки:

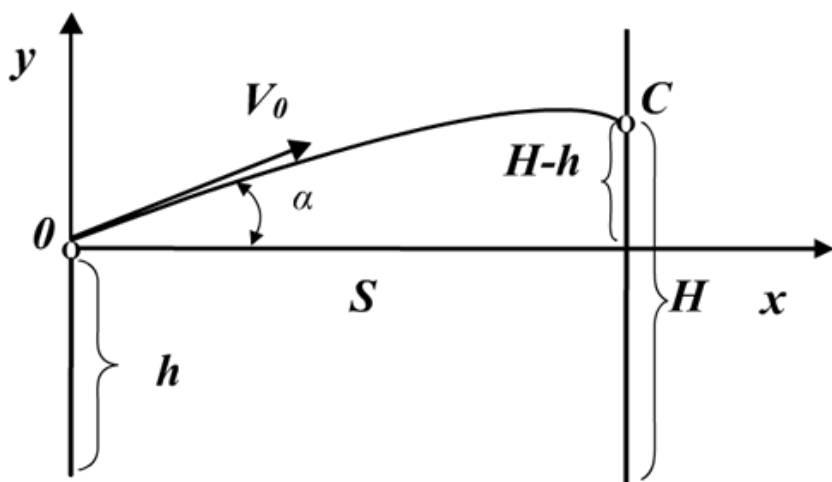


Положение щита с корзиной:



Попадание мяча в кольцо зависит от многих параметров: силы броска, угла полета мяча, расстояния до кольца, высоты кольца, высоты, с которой бросают мяч, массы мяча, сопротивления воздуха и др.

Приоритетными свойствами для моделирования будут начальная скорость мяча, угол, под которым брошен мяч, рост ученика. Будем считать мяч материальной точкой. Отметим на рисунке траекторию полета мяча:



Введем систему координат с началом отсчета в точке бросания мяча $x_0 = 0$, $y_0 = 0$.

Точка C – точка попадания мяча в корзину – лежит на максимальном расстоянии от точки O . Ее координаты $x = S$ и $y = H - h$.

Из кинематики известно, что координаты определяются из уравнений:

$$x = x_0 + S_x \quad y = y_0 + S_y$$

где $S_x = V_{0x}t + \frac{a_x t^2}{2}$, $S_y = V_{0y}t + \frac{a_y t^2}{2}$.

В нашем случае $a = g$, тогда

$$a_x = g_x = 0 \ (\vec{g} \perp OX) \text{ и } a_y = g_y = -g \ (\vec{g} \uparrow \downarrow OY)$$

Проекции V_0 на оси координат: $V_{0x} = V_0 \cos \alpha$; $V_{0y} = V_0 \sin \alpha$.

Следовательно,

$$S = V_0 t \cos \alpha, \quad (1)$$

$$H - h = V_0 t \sin \alpha - \frac{gt^2}{2}. \quad (2)$$

Из (1) находим время полета $t = \frac{S}{V_0 \cos \alpha}$ и подставляем в (2):

$$H - h = V_0 \sin \alpha \frac{S}{V_0 \cos \alpha} - g \frac{S^2}{V_0^2 \cos^2 \alpha}.$$

Получим квадратное уравнение относительно переменной S :

$$\frac{g}{V_0^2 \cos^2 \alpha} S^2 - \tan \alpha S + (H - h) = 0.$$

$$\text{Дискриминант } D = \tan^2 \alpha - \frac{4g(H - h)}{V_0^2 \cos^2 \alpha}.$$

Если $D < 0$, то корней у уравнения нет.

Если $D \geq 0$, то корни уравнения ищем по формулам:

$$S_1 = \frac{\tan \alpha + \sqrt{D}}{\frac{2g}{V_0^2 \cos^2 \alpha}}, \quad S_2 = \frac{\tan \alpha - \sqrt{D}}{\frac{2g}{V_0^2 \cos^2 \alpha}}.$$

Данное решение представлено в общем виде. Введем ограничения, в соответствии с правилами игры:

- угол броска не должен превышать 90° ;
- по правилам игры в баскетбол высота корзины $H = 3,05$ м;
- за ответ задачи будем принимать только расстояние, удовлетворяющее размерам баскетбольной площадки $0 < S < 28$;
- размеры мяча учитывать не будем.

Процесс решения будет включать следующее:

- объявление входных переменных:

- V_0 – начальная скорость мяча (м/с);
- A – угол (градусы);
- рост – рост школьника (м);

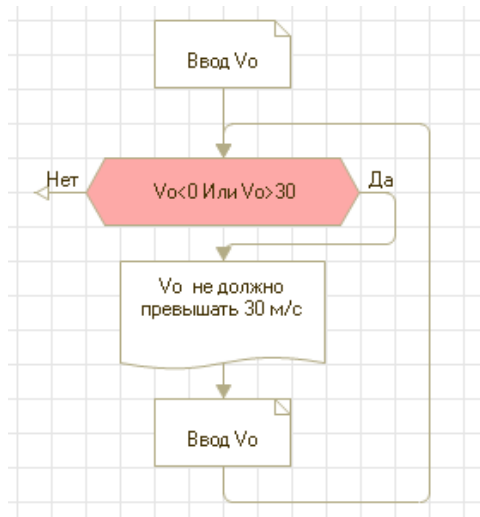
- ввод значений входных переменных при условиях:

- $0 < V_0 < 30$ (32,5 м/с – скорость ветра при урагане);
- $0 < A < 90$ (угол при броске);
- $1,20 < \text{Рост} < 2$ (рост школьника может колебаться в пределах от 1,20 до 2 м);

- вычисление расстояния;

- формулировка ответа с учетом размеров баскетбольной площадки.

Приведем фрагмент алгоритма для ввода входного значения V_0 :



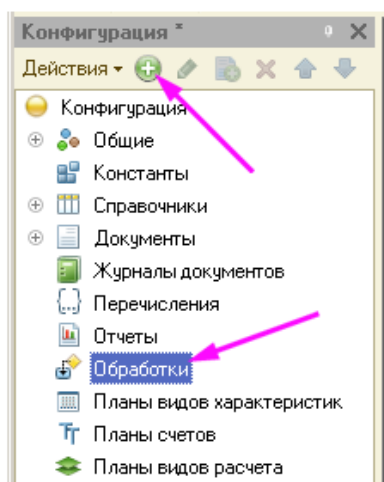
Аналогично следует организовать ввод угла и роста.

Для написания программы будем использовать объект конфигурации *Обработки*.

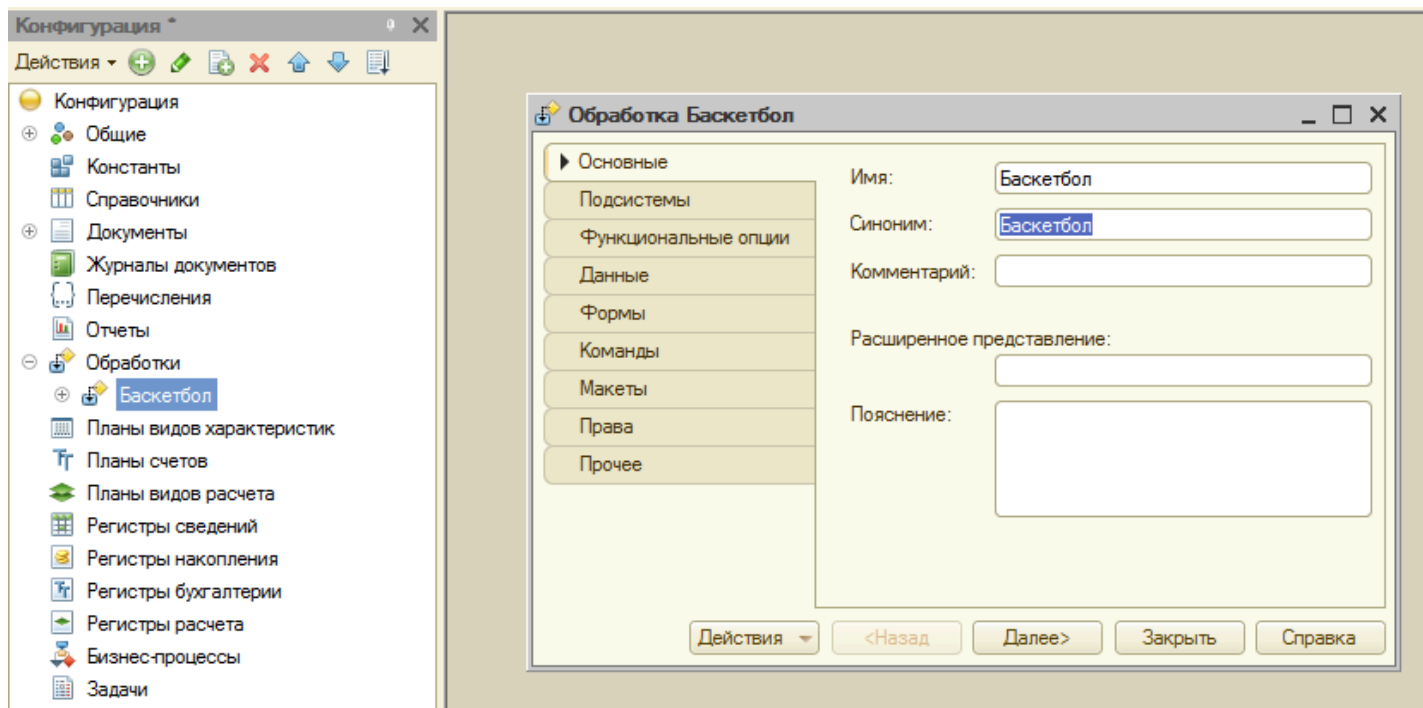
1. Запустим информационную базу в режиме *Конфигуратор*.

Настройки Конфигурации информационной системы

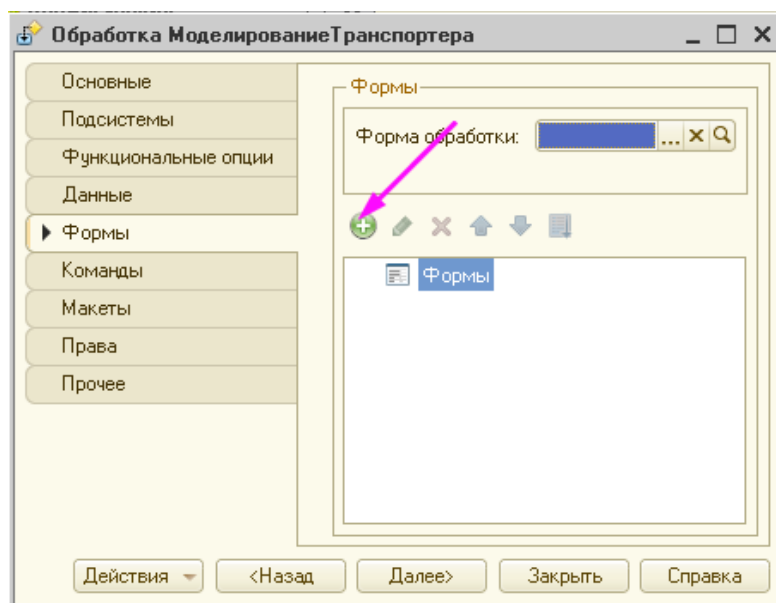
2. Выделим в дереве конфигурации *Обработки*, нажмем кнопку  *Добавить (Ins)*:




3. В появившемся окне обработки на вкладке *Основные* введем имя обработки – *Баскетбол*. При нажатии *Enter* автоматически будет заполнено свойство *Синоним*, которое представит имя в удобном для пользователя виде:




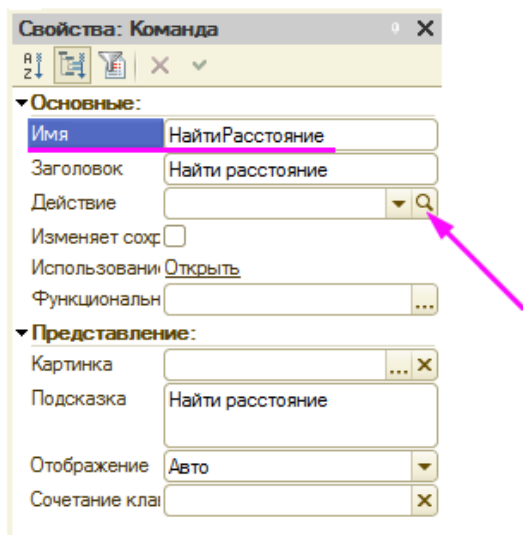
4. Перейдем на вкладку *Формы* –  *Добавить (Ins)*:



Данная форма будет предназначена для написания обработки (программы) и будет основной формой – *Готово*:

5. В окне формы на вкладке *Команды* добавим новую команду. Для этого перейдем на закладку *Команды* в правой верхней области окна и добавим с помощью кнопки  *Добавить* командной панели новую команду, как это показано на рисунке:

6. В окне свойств команды зададим ее имя – *НайтиРасстояние* (напомним, что в имени не ставятся пробелы, но каждое слово пишется с прописной буквы, в заголовке название появится автоматически в удобном для пользователя виде) и нажатием кнопки просмотра свойства  *Действие* создадим обработчик выполнения действия:



7. Откроется текстовый редактор разработки программного кода.

Поскольку мы попытались открыть несуществующий обработчик команды, программа создала его задав то же имя, что и у команды. Введем текст программы.

Объявим переменные:

```

НаКлиенте
Процедура НайтиРасстояние (Команда)
    Перец Vo;
    Перец A;
    Перец Рост;
    H = 3.05;

```

С помощью функции для ввода переменной типа Число вводим начальную скорость. С помощью цикла с предусловием может быть откорректировано введенное пользователем значение так, чтобы начальная скорость удовлетворяла условию $0 < V_0 < 30$:

```

ВвестиЧисло (Vo, "Введите начальную скорость мяча, м/с");
Пока Vo < 0 Или Vo > 30 Цикл
    Сообщить ("Скорость не должна превосходить 30 м/с");
    ВвестиЧисло (Vo, "Введите начальную скорость мяча");
КонецЦикла;

```

Далее вводим угол и рост:

```

ВвестиЧисло (A, "Введите угол броска, градусы");
Пока A < 0 Или A > 90 Цикл
    Сообщить ("Угол не долже превосходить 90 градусов");
    ВвестиЧисло (A, "Введите угол броска");
КонецЦикла;

ВвестиЧисло (Рост, "Введите рост школьника, м");
Пока Рост < 1 Или Рост > 2 Цикл
    Сообщить ("Рост школьника от 1,2 до 2 м");
    ВвестиЧисло (Рост, "Введите рост школьника");
КонецЦикла;

```

Начальные условия выводим с помощью функции Сообщить:

```

Сообщить ("Начальная скорость "+Vo+" м/с, угол "+A+" градусов,
| рост ученика "+Рост);

```

Переводим градусы в радианы:

```

A = A * 3.14 / 180;

```

Вычисляем дискриминант. Формулу $D = tg^2 \alpha - \frac{4g(H-h)}{V_0^2 \cos^2 \alpha}$ представляем в виде записи:

```
D = Pow(Tan(A), 2) - 4 * 9.8 * (H - Рост) / Pow(Vo * Cos(A), 2);
```

Проверяем условия существования корней квадратного уравнения и вычисляем корни:

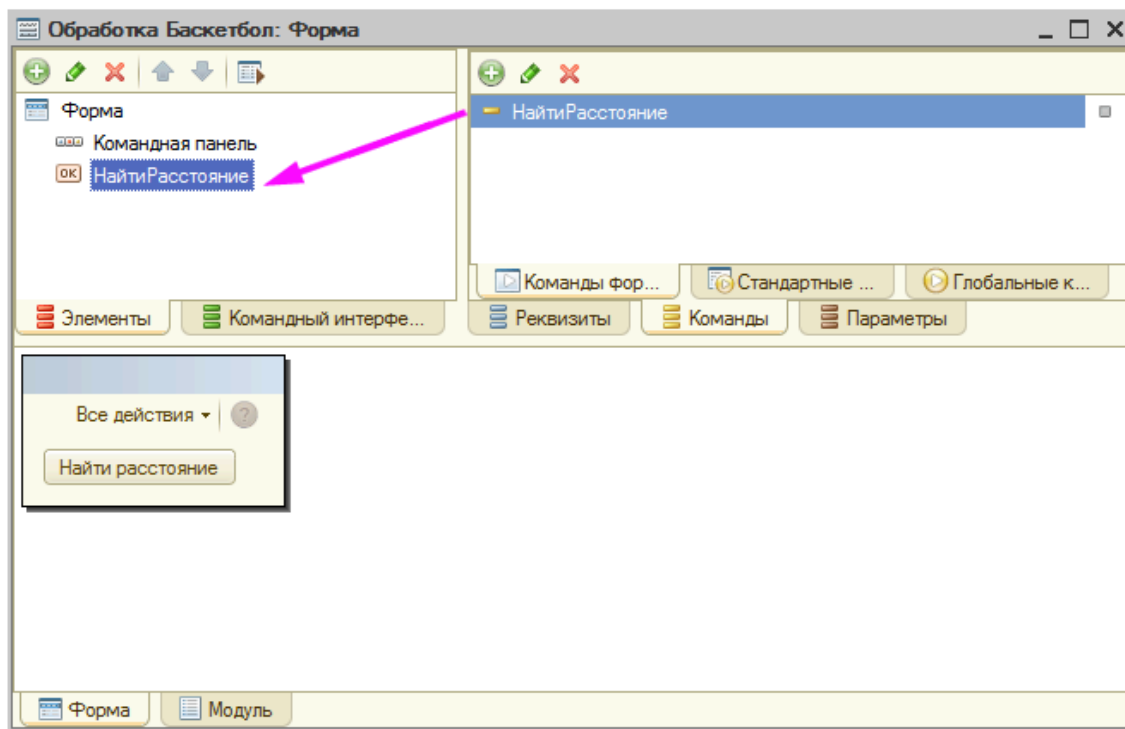
```
Если D >= 0 Тогда
    S1 = (Tan(A) + Sqrt(D)) / (2 * 9.8 / Pow(Vo * Cos(A), 2));
    S2 = (Tan(A) - Sqrt(D)) / (2 * 9.8 / Pow(Vo * Cos(A), 2));
Иначе
    S1 = 0;
    S2 = 0;
КонецЕсли;
```

Значения корней $S1$ и $S2$ нужны для того, чтобы сформулировать ответ:

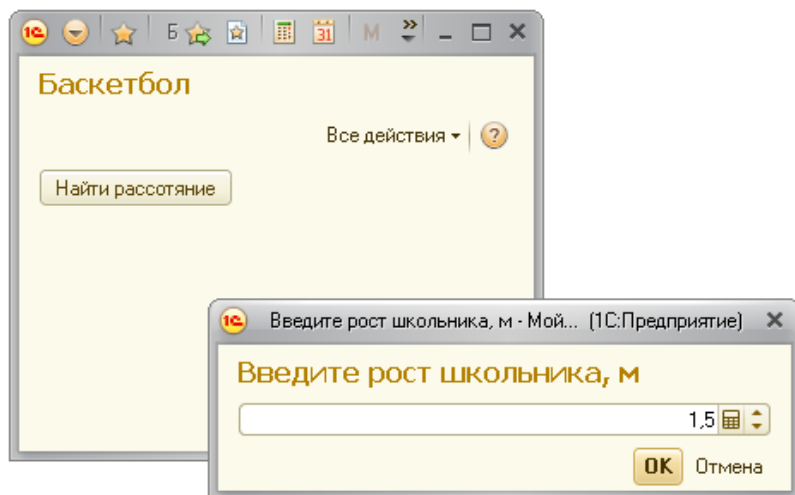
```
Если S1 > 0 И S1 < 28 И S1 <> S2 Тогда
    Сообщить ("Расстояние от корзины "+S1+" м");
ИначеЕсли S2 > 0 И S2 < 28 И S1 <> S2 Тогда
    Сообщить ("Расстояние от корзины "+S2+" м");
ИначеЕсли S2 > 0 И S2 < 28 И S1 = S2 Тогда
    Сообщить ("Расстояние от корзины "+S1+" м");
Иначе
    Сообщить ("При начальных условиях попадание невозможно");
КонецЕсли;
```

Скачать код программы

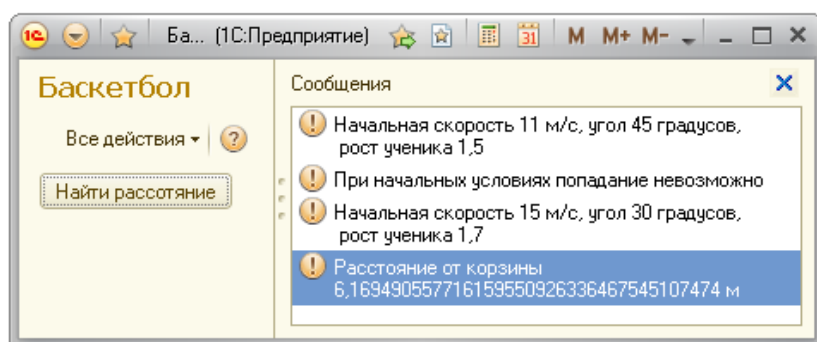
7. Перейдем на закладку *Форма* и перетащим мышью команду из правого верхнего окна (редактор команд) в левое верхнее окно (редактор элементов):



Запустим программу и введем исходные данные:



Проведем исследование, определим расстояние для нескольких значений роста школьников, начальной скорости и угла.



Задание 2

Усовершенствуйте модель, построенную задании 1. Введите дополнительные параметры для оценки размеров баскетбольного мяча и размеров корзины.

Как исправить синтаксические ошибки

Что делать, если программа зависла или требуется остановить ее из-за неправильного ввода данных?