

Муниципальное бюджетное общеобразовательное учреждение средняя
общеобразовательная школа №8 г. Тулы

Информатика и ИКТ
Практические работы по программированию
(9-11 классы)

Автор: Симонова Татьяна Николаевна

2012

Вступление

При изучении темы «Программирование» на уроках информатики часто возникает так называемая конфликтная ситуация: часть учащихся увлечена и с интересом выполняет задания, другая часть – с трудом понимает, а следовательно работает без интереса, не видя результата работы на уроке.



Рисунок 1

В своей разработке я предлагаю серию практических работ, при выполнении которых решается *одна задача по технологии «от простого к сложному»*, что позволяет избежать этого конфликта.

На протяжении всех работ учащиеся строят, совершенствуют, исследуют модель движения снаряда (Н.Д. Угринович, Информатика и ИКТ, профильный уровень, 11 класс).

Автор учебника предлагает готовую программу. Методика проведения предлагаемых практических работ, формулировка заданий направлены, в основном, на самостоятельный поиск решений.

Важно!

Учащиеся должны не только увидеть проблему, но и предложить пути решения.

Задача учителя: совместно с учащимися обсудить, «проговорить» алгоритм решения, а затем предложить воспользоваться справочным материалом среды программирования или Интернет-ресурсами.

Цель проведения практических работ:

- развитие познавательного интереса;
- отработка ранее изученных алгоритмов;
- изучение новых возможностей языка программирования PascalABC;
- формирование универсальных учебных действий, связанных с поиском информации, необходимой для решения поставленной задачи.

Рекомендуемый возраст: 9-11 классы (зависит от профиля обучения).

На момент выполнения первой практической работы учащиеся знают:

- основные алгоритмические структуры (линейная, ветвление, цикл с параметром);
- операторы языка PascalABC (ввод с экрана, вывод на экран, присваивание, условный оператор, цикл с параметром)
- структуру программы.

Опыт решения задач:

- простые задачи на циклы и ветвления.

Тексты программ приведены на языке PascalABC.

Время выполнения: 20-40 минут на каждую работу (зависит от уровня знаний).

Оценивание работ: по конечному результату.

Содержание задачи

Формулируем единую задачу для всей серии практических работ:

Из пушки произвести выстрел по заданной цели. Провести анализ:

1. Определить разброс по углу пуска при заданной начальной скорости снаряда.
2. Определить разброс по скорости при заданном угле пуска.

Требования к модели :

- Наглядный и удобный вывод результатов.

Допущения:

- Сопротивлением воздуха можно пренебречь.
- Ускорение свободного падения считаем постоянным.
- Модель работает только в вертикальной плоскости.

Описательная модель:

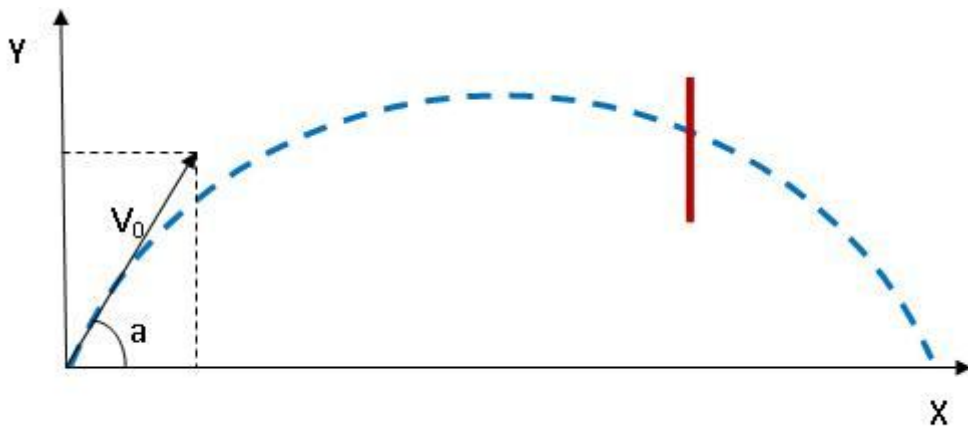


Рисунок 2

$$X=V_0 \cdot \cos(a) \cdot t;$$

$$Y=V_0 \cdot \sin(a) \cdot t - gt^2/2;$$

Практическая работа №1

Задача: вывести на экран траекторию движения снаряда при следующих исходных данных:

Начальная скорость = 80 м/с

Угол пуска = 60 градусов

Время полета = 20 сек.

Решение:

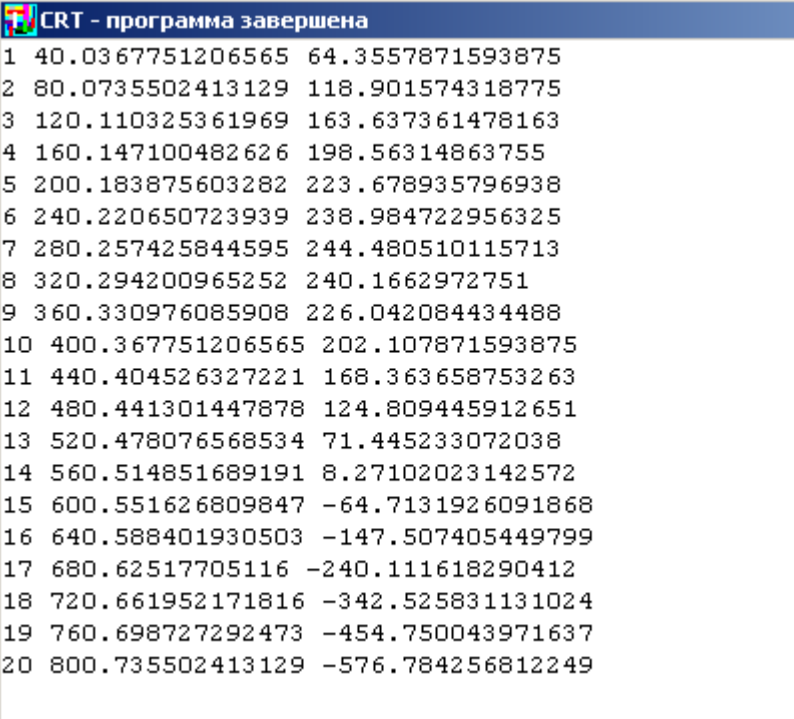
Программа	Результат выполнения программы
<pre>program s1; uses crt; var a,v,x,y:real; i,t:integer; const g=9.81; begin v:=80; a:=60*3.14/180; for t:=1 to 20 do begin x:=v*t*cos(a); y:=v*t*sin(a)-g*t*t/2; writeln(t,' ',x,' ',y); end; end.</pre>	 <p>1 40.0367751206565 64.3557871593875 2 80.0735502413129 118.901574318775 3 120.110325361969 163.637361478163 4 160.147100482626 198.56314863755 5 200.183875603282 223.678935796938 6 240.220650723939 238.984722956325 7 280.257425844595 244.480510115713 8 320.294200965252 240.1662972751 9 360.330976085908 226.042084434488 10 400.367751206565 202.107871593875 11 440.404526327221 168.363658753263 12 480.441301447878 124.809445912651 13 520.478076568534 71.445233072038 14 560.514851689191 8.27102023142572 15 600.551626809847 -64.7131926091868 16 640.588401930503 -147.507405449799 17 680.62517705116 -240.111618290412 18 720.661952171816 -342.525831131024 19 760.698727292473 -454.750043971637 20 800.735502413129 -576.784256812249</p>

Рисунок 3

Практическая работа №2

Проанализируем результат выполнения программы.

Недостатки:

1. Ненаглядный вывод результатов («плавающие» колонки, много знаков после запятой).
2. Координата по оси «у» становится отрицательной (снаряд уходит под землю).

Задача: Устранить недостатки, используя форматный вывод и цикл «до» (оператор while).

Решение:

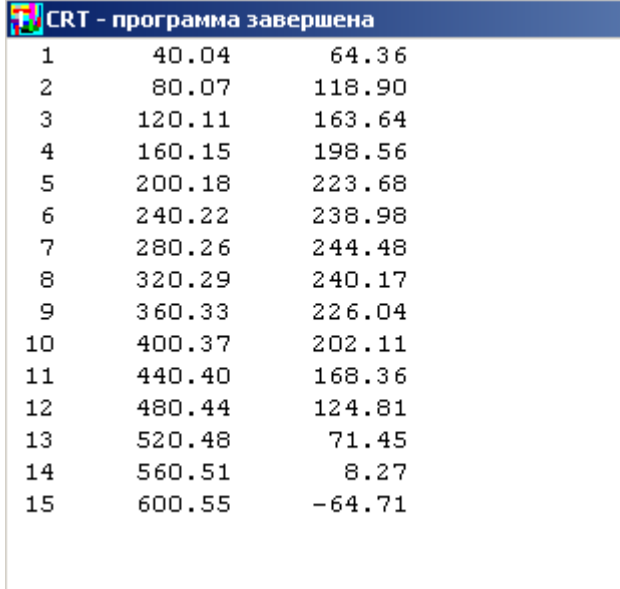
Программа	Результат выполнения программы																																													
<pre>program s1; uses crt; var a,v,x,y,t:real; i:integer; const g=9.81; begin v:=80; a:=60*3.14/180; t:=0; while y>=0 do begin t:=t+1; x:=v*t*cos(a); y:=v*t*sin(a)-g*t*t/2; writeln(t:3:0,' ',x:10:2,' ',y:10:2); end; end.</pre>	 <p>CRT - программа завершена</p> <table><tbody><tr><td>1</td><td>40.04</td><td>64.36</td></tr><tr><td>2</td><td>80.07</td><td>118.90</td></tr><tr><td>3</td><td>120.11</td><td>163.64</td></tr><tr><td>4</td><td>160.15</td><td>198.56</td></tr><tr><td>5</td><td>200.18</td><td>223.68</td></tr><tr><td>6</td><td>240.22</td><td>238.98</td></tr><tr><td>7</td><td>280.26</td><td>244.48</td></tr><tr><td>8</td><td>320.29</td><td>240.17</td></tr><tr><td>9</td><td>360.33</td><td>226.04</td></tr><tr><td>10</td><td>400.37</td><td>202.11</td></tr><tr><td>11</td><td>440.40</td><td>168.36</td></tr><tr><td>12</td><td>480.44</td><td>124.81</td></tr><tr><td>13</td><td>520.48</td><td>71.45</td></tr><tr><td>14</td><td>560.51</td><td>8.27</td></tr><tr><td>15</td><td>600.55</td><td>-64.71</td></tr></tbody></table>	1	40.04	64.36	2	80.07	118.90	3	120.11	163.64	4	160.15	198.56	5	200.18	223.68	6	240.22	238.98	7	280.26	244.48	8	320.29	240.17	9	360.33	226.04	10	400.37	202.11	11	440.40	168.36	12	480.44	124.81	13	520.48	71.45	14	560.51	8.27	15	600.55	-64.71
1	40.04	64.36																																												
2	80.07	118.90																																												
3	120.11	163.64																																												
4	160.15	198.56																																												
5	200.18	223.68																																												
6	240.22	238.98																																												
7	280.26	244.48																																												
8	320.29	240.17																																												
9	360.33	226.04																																												
10	400.37	202.11																																												
11	440.40	168.36																																												
12	480.44	124.81																																												
13	520.48	71.45																																												
14	560.51	8.27																																												
15	600.55	-64.71																																												

Рисунок 4

Практическая работа №3.

Проанализируем результат выполнения программы:

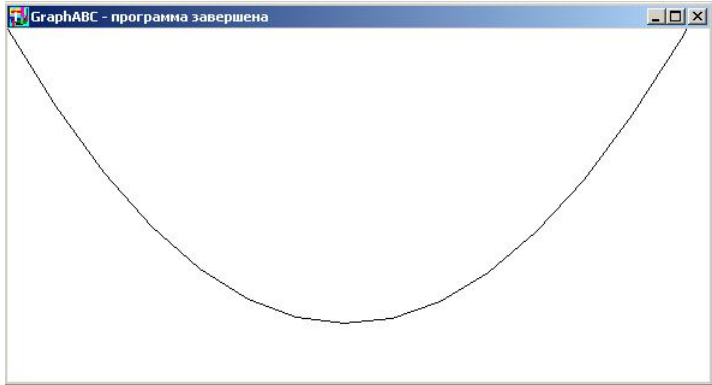
Вывод результатов удовлетворителен, таблица наглядна. Однако возникает желание визуализировать траекторию движения.

Задача: нарисовать траекторию движения снаряда.

Действия:

- Подключить графический модуль
- В цикл добавить оператор, последовательно соединяющий точки (X,Y).

Решение:

Программа	Результат выполнения программы
<pre>program s1; uses GraphABC; var a,v,x,y,t:real; i:integer; const g=9.81; begin v:=80; a:=60*3.14/180; t:=0; MoveTo(0,0); while y>=0 do begin t:=t+1; x:=v*t*cos(a); y:=v*t*sin(a)-g*t*t/2; writeln(t:5:1,' ',x:10:2,' ',y:10:2); LineTo(round(x),round(y)); end; end.</pre>	<p>Таблица осталась без изменений.</p>  <p>Рисунок 5</p>

PS. Скорее всего учащиеся не обратят сразу внимание на тип параметров процедуры LineTo(x,y:integer) и не поставят округление (round(x),round(y)). Тем лучше! Одной проблемой больше, т.к. задача решаться не будет.

Практическая работа №4.

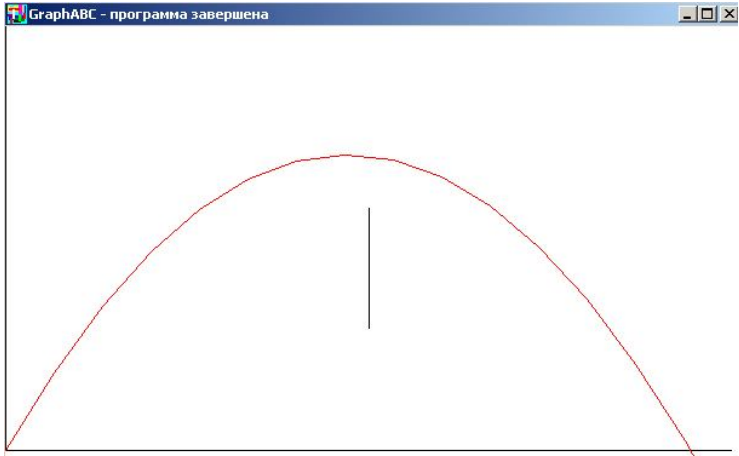
Анализируем результат выполнения предыдущей программы.

Недостаток очевиден: график перевернут (вертикальная ось направлена вниз).

Задача:

- внести изменения в программу, которые приведут к нормальному отображению траектории (подсказка: новых операторов не потребуется, достаточно выполнить симметричные преобразования и сместить график вниз);
- нарисовать неподвижную цель с заданными параметрами в воздухе (вертикальную линию над уровнем «земли») и оси координат;
- траекторию снаряда изобразить красным цветом.

Решение:

Программа	Результат выполнения программы
<pre>program s1; uses GraphABC; var a,v,x,y,t:real; i:integer; const g=9.81; begin v:=80; a:=60*3.14/180; t:=0; Line(0,0,0,350); LineTo(600,350); Line(300,150,300,250); SetPenColor(clRed); MoveTo(0,350); while y>=0 do begin t:=t+1; x:=v*t*cos(a); y:=v*t*sin(a)-g*t*t/2; writeln(t:5:1,',',x:10:2,',',y:10:2); LineTo(round(x),round(350-y)); end; end.</pre>	<p>Таблица осталась без изменений.</p>  <p>Рисунок 6</p>

Практическая работа №5.

Обратимся к *общей постановке задачи*:

Из пушки произвести выстрел по заданной цели. Провести анализ:

1. Определить разброс по углу пуска при заданной начальной скорости снаряда.
2. Определить разброс по скорости при заданном угле пуска.

Уточняем задачу:

- разброс по углу пуска должен определяться в результате одного запуска программы
- разброс по начальной скорости должен определяться в результате одного запуска программы


Решение:

В программе используем структуру вложенных циклов:

во внешнем цикле вычисляется новый угол (скорость), во внутреннем рассчитывается траектория.

Шаг изменения угла (скорости) подбирается в процессе отладки программы.

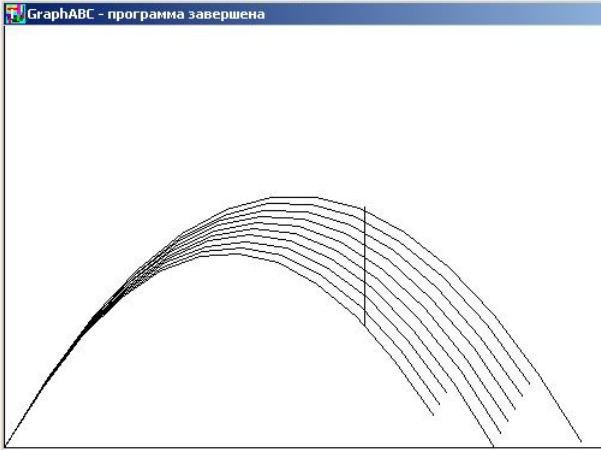
Определение разброса угла пуска.

Программа	Результат выполнения программы
<pre>program s5; uses GraphABC; var a,da,v,x,y:real; i,j,t:integer; const g=9.81; begin v:=80; a:=33*3.14/180; da:=2*3.14/180; Line(0,0,0,350); LineTo(600,350); Line(300,150,300,250); writeln('начальная скорость ',v); For i:=1 to 10 do begin t:=0; a:=a+da; x:=0; y:=0; MoveTo(0,350); while y>=0 do begin LineTo(round(x),round(350-y)); t:=t+1; x:=v*t*cos(a);</pre>	 <p>Рисунки 7, 8</p>

<pre> y:=v*t*sin(a)-g*t*t/2; end; writeln(i,' угол пуска',a*180/3.14:10:2); end; end. </pre>	
-----------------------------------------------------------------------------------------------	--

Вывод: при углах пуска, лежащих в диапазоне от 35 до 51 градусов, и начальной скорости 80 м/с цель будет поражена.

Определение разброса начальной скорости.

Программа	Результат выполнения программы																						
<pre> program s5; uses GraphABC; var a,da,v,x,y:real; i,j,t:integer; const g=9.81; begin v:=64; a:=60*3.14/180; da:=2*3.14/180; Line(0,0,0,350); LineTo(600,350); Line(300,150,300,250); writeln('угол пуска ',a*180/3.14:10:2); For i:=1 to 10 do begin t:=0; v:=v+1; x:=0; y:=0; MoveTo(0,350); while y>=0 do begin LineTo(round(x),round(350-y)); t:=t+1; x:=v*t*cos(a); y:=v*t*sin(a)-g*t*t/2; end; writeln(i,' скорость ',v:10:2); end; end. </pre>	 <table border="1"> <tr> <td>угол пуска</td> <td>60.00</td> </tr> <tr> <td>1 скорость</td> <td>65.00</td> </tr> <tr> <td>2 скорость</td> <td>66.00</td> </tr> <tr> <td>3 скорость</td> <td>67.00</td> </tr> <tr> <td>4 скорость</td> <td>68.00</td> </tr> <tr> <td>5 скорость</td> <td>69.00</td> </tr> <tr> <td>6 скорость</td> <td>70.00</td> </tr> <tr> <td>7 скорость</td> <td>71.00</td> </tr> <tr> <td>8 скорость</td> <td>72.00</td> </tr> <tr> <td>9 скорость</td> <td>73.00</td> </tr> <tr> <td>10 скорость</td> <td>74.00</td> </tr> </table> <p style="text-align: center;">Рисунки 9,10</p>	угол пуска	60.00	1 скорость	65.00	2 скорость	66.00	3 скорость	67.00	4 скорость	68.00	5 скорость	69.00	6 скорость	70.00	7 скорость	71.00	8 скорость	72.00	9 скорость	73.00	10 скорость	74.00
угол пуска	60.00																						
1 скорость	65.00																						
2 скорость	66.00																						
3 скорость	67.00																						
4 скорость	68.00																						
5 скорость	69.00																						
6 скорость	70.00																						
7 скорость	71.00																						
8 скорость	72.00																						
9 скорость	73.00																						
10 скорость	74.00																						

Вывод: при скорости, лежащей в диапазоне от 65 до 74 м/с, и начальном угле пуска 60 градусов цель будет поражена.

Домашняя практическая работа:

- Задайте для каждой траектории «случайный цвет».
- Сформулируйте задание (ряд вопросов), которое можно выполнить, используя построенную модель.
- *Проведите моделирование : ответьте на свои вопросы.

Список источников:

Н. Д. Угринович. Информатика и ИКТ. Профильный уровень: учебник для 11 класса. М.: БИНОМ. Лаборатория знаний, 2009.

Н.И. АМЕЛИНА, А.А. ЧЕКУЛАЕВА, М.И. ЧЕРДЫНЦЕВА. ГРАФИКА В СИСТЕМЕ PascalABC. Федеральное государственное образовательное учреждение высшего профессионального образования «ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ», 2008.