

Некоторые методические аспекты использования графических возможностей среды Pascal ABC в школьном курсе математики

©2016 Агаханов С. А.^{1,3}, Рагимханова Г. С.^{1,3},
Рагимханова А. Р.², Агаханова А. А.¹

¹ Дагестанский государственный педагогический университет,
Махачкала, Россия; e-mail: gulnara_6789@mail.ru

² Дагестанский государственный университет,
Махачкала, Россия; e-mail: gulnara_6789@mail.ru

³ Дагестанский институт развития образования,
Махачкала, Россия; e-mail: gulnara_6789@mail.ru

РЕЗЮМЕ. Цель. Цель данного исследования заключается в определении графических возможностей среды программирования в школьном курсе математики. **Методы.** Методы компьютерной графики в учебном процессе. **Результаты.** В качестве практической реализации заданной цели разработана программа на языке Pascal ABC для построения графиков функций. **Выводы.** Разработанная программа на языке Pascal ABC дает возможность получить эскизы графиков различных функций.

Ключевые слова: Pascal ABC, график функции, система координат.

Формат цитирования: Агаханов С. А., Рагимханова Г. С., Рагимханова А. Р., Агаханова А. А. Некоторые методические аспекты использования графических возможностей среды Pascal ABC в школьном курсе математики // Известия Дагестанского государственного педагогического университета. Психолого-педагогические науки. 2016. Т. 10. № 4. С. 28-32.

Some Methodological Aspects of Using the Pascal Environment Graphic Possibilities in School-Time Math Tuition

©2016 Selimkhan A. Agakhanov^{1,3}, Gulnara S. Ragimkhanova^{1,3},
Alina R. Ragimkhanova², Aida A. Agakhanova¹

¹ Dagestan State Pedagogical University,
Makhachkala, Russia; e-mail: gulnara_6789@mail.ru

² Dagestan State University,
Makhachkala, Russia; e-mail: gulnara_6789@mail.ru

³ Dagestan Institute of Education Development,
Makhachkala, Russia; e-mail: gulnara_6789@mail.ru

ABSTRACT. Aim. The aim of this research is to define graphic possibilities of programming environment in school-time math tuition. **Methods.** Methods of computer graphics in the learning process. **Results.** In the function of specified objective practical realization, the authors of the article develop Pascal ABC program for producing the graphics of functions. **Conclusions.** Developed Pascal ABC program gives an opportunity to get thumbnails of different functions graphics.

Keywords: Pascal ABC, graphic of function, coordinate system.

For citation: Agakhanov S. A., Ragimkhanova G. S., Ragimkhanova A. R., Agakhanova A. A. Some Methodological Aspects of Using the Pascal Environment Graphic Possibilities in School-Time Math Tuition. Dagestan State Pedagogical University. Journal. Pedagogical and Psychological Sciences. 2016. Vol. 10. No. 4. Pp. 28-32. (In Russian)

Введение

По словам доктора философии Айвена Сазерленда: «Дисплей, подключенный к ЭВМ, представляется мне окном в Алисину Страну чудес, где программист может изображать любые объекты, описываемые хорошо известными законами природы, либо чисто воображаемые объекты, подчиняющиеся законам, записанным в программе».

Из сказанного следует, какую важную роль может сыграть применение компьютерной графики в учебном процессе как в школе, так и в вузе. Этим можно установить и межпредметные связи при изучении различных предметов.

На сегодня существует множество прикладных программ (Excel, Mathcad, Mathematic и т. д.), которые можно использовать для наглядного представления изучаемых процессов [2]. Тем не менее, как происходит процесс построения графика и его изображение на экране дисплея занимает особое место в работе учителя в школе или преподавателя в вузе.

Цель и методы исследования

Здесь мы хотим остановиться на применении графических возможностей языка программирования Pascal ABC. Для работы в графическом режиме Pascal ABC следует командой `uses Graph` ABC подключить графический модуль. Графический экран Pascal ABC представляет собой растр. Растр – это равномерная сетка из пикселей, упорядоченных по строкам и столбцам. Под растровым изображением понимают набор цветных пикселей, расположенных специальным образом на равномерной сетке. Для примера, растр можно создать на бумажном листе, расчертив его вертикальными и горизонтальными линиями так, чтобы получилась равномерная сетка (все ячейки сетки были одинаковые по размерам квадраты). Покрасив соответствующим образом ячейки сетки цветом, несовпадающим с цветом бумаги, получим растровые изображения. Одной из характеристик растра является его разрешающая способность, т. е. количество строк и столбцов из пикселей в графическом

режиме. Разрешающая способность нашего экрана $m \times n$, где m – строки и n – столбцы.

Наряду с пикселями вводим такие понятия, как точки на экране и координаты точек на экране. Для этого рассмотрим прямоугольную декартову систему координат XO_1Y_1 , где начало координат O_1 расположено в верхнем левом углу монитора, ось O_1X_1 направлена слева направо, а ось O_1Y_1 сверху вниз. Систему координат XO_1Y_1 назовем экранной системой координат. Здесь $0 \leq X_1 \leq n$, $0 \leq Y_1 \leq m$.

Привычную для нас прямоугольную декартову систему координат, обозначим через XOY и назовем эту систему истинной системой координат.

X_0, Y_0 – экранные координаты для начала координат истинной системы.

Нетрудно заметить, что $0 \leq X_0 \leq n$, $0 \leq Y_0 \leq m$.

Чтобы график на экране выглядел в привычной форме, установим связь между экранными и истинными координатами.

Экранные координаты X_1, Y_1 и истинные координаты X, Y связаны следующими соотношениями:
 $-X_0 \leq X \leq n - X_0$, $-(m - Y_0) \leq Y \leq Y_0$.

Далее, вводим два положительных числа k_x, k_y , которые зависят от разрешающей способности экрана и предназначены для сжатия ($k_x < 1, k_y < 1$) и растяжения ($k_x > 1, k_y > 1$) множества значений аргумента X и функции Y по осям координат.

Чтобы точки графика изображались на экране дисплея необходимо переходить от истинных координат к экранным координатам.

Связь между экранными координатами X_1, Y_1 и истинными координатами X, Y задается формулами:

$$X_1 = X + X_0; Y_1 = -Y + Y_0.$$

При работе в графическом режиме используются следующие графические процедуры:

PutPixel (x, y, color) – закрашивает точку с графическими координатами (x, y) цветом color. Цвет можно указывать как числом, так и предопределенной константой. Список этих констант и кодов для 16-цветного режима приведен в таблице.

GetMaxX, GetMaxY – возвращают наибольшие значения графических координат по осям абсцисс и ординат соответственно для выбранного графического режима. Например, для разрешения 800*600 эти значения будут равны 799 и 599.

SetColor (color) – устанавливает текущий цвет. Все нарисованное вызванными после обращения к SetColor подпрограммами будет иметь цвет color (до нового вызова процедуры SetColor).

Таблица

**Константы и коды
для 16-цветного режима**

Цвет	Константа	Код
Черный	Black	0
Синий	Blue	1
Зеленый	Green	2
Бирюзовый	Cyan	3
Красный	Red	4
Розовый	Magenta	5
Коричневый	Brown	6
Светло-серый	LightGray	7
Темно-серый	DarkGray	8
Светло-синий	LightBlue	9
Светло-зеленый	LightGreen	10
Светло-бирюзовый	LightCyan	11
Светло-красный	LightRed	12
Светло-розовый	LightMagenta	13
Желтый	Yellow	14
Белый	White	15

Исключение составляет процедура PutPixel, которая выполняет закрашивание точки в соответствии со своим собственным параметром color.

SetBkColor (color) – устанавливает текущий цвет фона экрана.

ClearDevice – очищает весь экран, закрашивая его текущим цветом фона.

Рисование графических примитивов и фигур.

Процедура SetLineStyle (style, def, thickness) – устанавливает стиль рисуемых линий. Стилем линии называют ее внешний вид. Линия может быть сплошной, тогда аргумент style принимает значение Solidln,

пунктирной (Dottedln), штрихпунктирной (Centerln), штриховой (Dashedln). Стиль может быть задан и программистом (UserBitLn). Аргумент def используется при задании своего стиля линии (1 – нормальная толщина, 3 – толстая линия).

Процедура Line (x1, y1, x2, y2) рисует отрезок прямой линии, проведенной между точками с графическими координатами (x1, y1) и (x2, y2).

Circle (x, y, R) выводит окружность с центром в точке (x, y) радиусом R.

Arc (x, y, beg, end, R) рисует дугу с центром окружности дуги в точке (x, y). Параметры beg и end – это начальный и конечный углы, отсчитываемые от горизонтальной оси против часовой стрелки в градусах, R – радиус окружности.

Rectangle (x1, y1, x2, y2) – вывод прямоугольника с координатами левого верхнего угла (x1, y1) и правого нижнего (x2, y2).

Некоторые подпрограммы используют при своей работе в качестве начальной позиции для вывода графический указатель. Этот указатель на экране не отображается. Устанавливает указатель в точку с координатами (x, y) процедура MoveTo (x, y). Используется курсор в основном следующими подпрограммами:

– LineTo (x, y) – рисует отрезок с началом в точке текущего положения графического курсора и концом в точке (x, y). При этом курсор перемещается в точку (x, y). Эта процедура очень удобна для построения сложных ломаных линий.

– OutText ('текст') – выводит указанный текст, начиная с текущего положения графического указателя так, что положение указателя совпадает с левым верхним углом первого выводимого символа.

– OutTextXY (x, y, 'текст') – выводит текст, начиная с точки (x, y). Способ вывода текста задается процедурой SetTextStyle.

Результаты

Хотелось бы здесь отметить, что единицы масштаба по осям координат в системе координат XOY обычно не совпадают, это и необязательно, ибо мы строим эскиз графика [1]. Ниже представлена программа, разработанная в среде Pascal ABC, для графика функции $y = \cos x$ на отрезке $[-\pi; \pi]$.

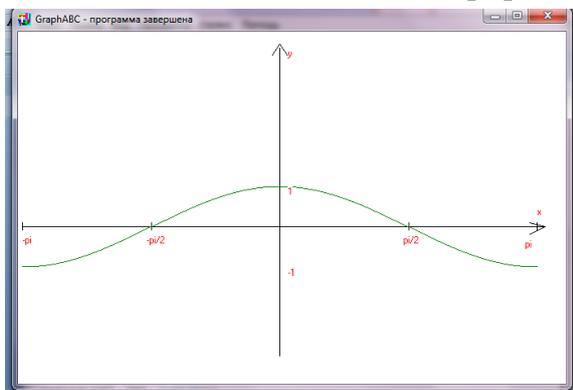
```
Uses graph ABC;
var x,y,x0,y0,kx,ky:integer;
begin
```

```

line (5, 240, 645, 240);
line (320, 20, 320, 400);
setfontcolor (clred);
textout (635, 215, 'x');
textout (330, 20, 'y');
x0:=320;
y0:=240;
ky:=50;
for x:=-315 to 315 do
begin
y:=trunc (ky*cos(x/(2*ky)));
putpixel (x0+ x, y0- y, clgreen);
end;
line (635, 235, 635, 245);
line (5, 235, 5, 245);
textout (5, 250, '-pi');
textout (620, 255, 'pi');
line (163, 235, 163, 245);
line (478, 235, 478, 245);
textout (157, 250, '-pi/2');
textout (471, 250, 'pi/2');
line (320, 15, 310, 30);
line (320, 15, 330, 30);
line (645, 240, 625, 235);
line (645, 240, 625, 250);
textout (330, 240-ky, '1');
textout (330, 240+ky, '-1');
end. (см. график 1).

```

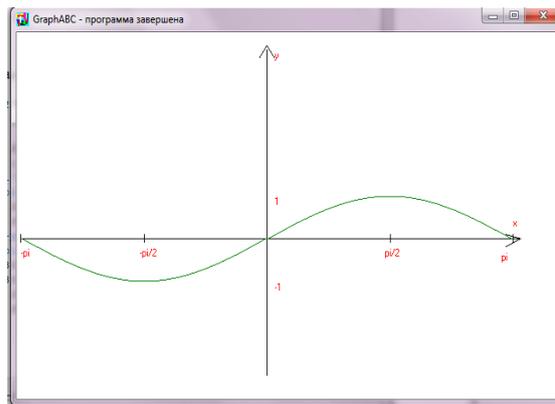
График 1



Выводы

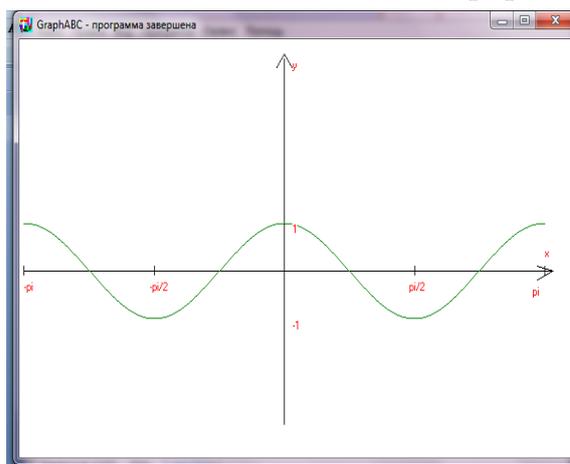
Заменяя в строке программы $y := \text{trunc}(ky * \cos(x/(2 * ky)))$ функцию \cos на \sin получим график функции $y = \sin x$ на отрезке $[-\pi; \pi]$ (см. график 2)

График 2



Заменяя в строке программы $y := \text{trunc}(ky * \cos(x/(2 * ky)))$ аргумент $x/(2 * ky)$ функции \cos на аргумент $2 * x/(2 * ky)$ получим график функции $y = \cos 2x$. (график 3).

График 3



Заменяя в строке программы функцию \cos на другие тригонометрические функции и, выбирая kx , ky учитывая их области определения и множества значений, и разрешающую способность экрана, можно получить графики и этих функций. Используя данную методику можно построить и графики других элементарных функций.

Литература

1. Мальхина М. П. Программирование на языке высокого

уровня Turbo Pascal. Санкт-Петербург, 2006. 544 с.

2. Немнюгин С. А., Перколаб Л. В. Изучаем Turbo Pascal. Санкт-Петербург, 2003. 320 с.

3. Эсетов Ф. Э., Эсетов М. Ф., Юзбекова М. Н. Подготовка учителя

математики на основе информационных технологий // Научный журнал: научные исследования и образование. 2015. № 19. С. 333-336.

References

1. Malykhina M. P. *Programmirovaniye na yazyke vysokogo urovnya Turbo Pascal* [Programming in Turbo Pascal high level language]. Saint Petersburg, 2006, 544 p. (In Russian)

2. Nemnyugin S. A., Perkolab L. V. *Izuchaem Turbo Pascal* [Studying Turbo Pascal]. Saint Petersburg, 2003, 320 p. (In Russian)

СВЕДЕНИЯ ОБ АВТОРЕ

Принадлежность к организации

Агаханов Селимхан Агаханович, кандидат физико-математических наук, доцент кафедры информатики и вычислительной техники (ИВТ), факультет математики, физики и информатики (МФИ) Дагестанский государственный педагогический университет (ДГПУ), Дагестанский институт развития образования (ДИРО), Махачкала, Россия; e-mail: gulnara_6789@mail.ru

Рагимханова Гюльнара Сарухановна, кандидат физико-математических наук, доцент кафедры ИВТ, заместитель декана по науке и информатизации факультета МФИ, ДГПУ; ДИРО, Махачкала, Россия; e-mail: gulnara_6789@mail.ru

Рагимханова Алина Романовна, студентка 4-го курса факультета информатики и информационных технологий, Дагестанский государственный университет (ДГУ), Махачкала, Россия; e-mail: gulnara_6789@mail.ru

Агаханова Аида Арсеновна, студентка 3-го курса факультета МФИ, ДГПУ, Махачкала, Россия; e-mail: gulnara_6789@mail.ru

Принята в печать 24.08.2016 г.

3. Esetov F. E., Esetov M. F., Yuzbekova M. N. Mathematics teacher training based on information technologies. *Nauchnyy zhurnal: nauchnye issledovaniya i obrazovanie* [Scientific journal: scientific researches and education]. 2015. No. 19. Pp. 333-336. (In Russian)

INFORMATION ABOUT AUTHORS

Affiliations

Selimhan A. Agakhanov, Ph. D. (Physics and Math), assistant professor, the chair of Computer Science (SC), the faculty of Math, Physics and Computer Science (MPCS), Dagestan State Pedagogical University (DSPU); Dagestan Institute of Education Development (DIED), Makhachkala, Russia; e-mail: gulnara_6789@mail.ru

Gulnara S. Ragimkhanova, Ph. D. (Physics and Math), assistant professor, the chair of CS, deputy of the science and informational support dean, the faculty of MPCS, DSPU; DIED, Makhachkala, Russia; e-mail: gulnara_6789@mail.ru

Alina R. Ragimkhanova, 4th year student, the faculty of Computer Science and Information Technologies, Dagestan State University (DSU), Makhachkala, Russia; e-mail: gulnara_6789@mail.ru

Aida A. Agakhanova, 3rd year student, the faculty of MPCS, DSPU, Makhachkala, Russia; e-mail: gulnara_6789@mail.ru

Received 24.08.2016.