

DOI: 10.32517/2221-1993-2022-21-6-30-36

С. С. Гербут

Лицей № 2, г. Тула, Россия

ПРОПЕДЕВТИКА ПРОГРАММИРОВАНИЯ: КАК ПОДГОТОВИТЬ УЧАЩИХСЯ К ВОСПРИЯТИЮ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО ПРОГРАММИРОВАНИЯ*

Аннотация

В статье описаны методические основы подготовки к программированию в основной школе и, в частности, к восприятию объектно-ориентированного программирования в будущем. Данную методику можно реализовать в курсе внеурочной деятельности или воспользоваться ее фрагментами для реализации на уроке. Программирование — один из важнейших разделов в базовом школьном курсе информатики. Это интересный и одновременно довольно сложный для начального обучения раздел. С учетом того, что базовый курс информатики начинается с седьмого класса и составляет всего один час в неделю (углубленный — два часа в неделю), на программирование выделяется время только в восьмом классе и не так уж много. А познакомить детей нужно с большим числом понятий, научить писать программы — от линейных до обработки массивов. В связи с этим предлагается использовать пропедевтический курс внеурочной деятельности для учащихся шестых-седьмых классов с целью их знакомства с основами структурного программирования на примере работы с исполнителями, позволяющими реализовать знакомство со структурой программы, синтаксическими требованиями, основными алгоритмическими конструкциями через рисование и движение графических объектов (на выбор учителя — на алгоритмическом языке, PascalABC, Python, C++). Методические разработки, описанные в статье, можно использовать для тех исполнителей, которых не будет на уроках, или, наоборот, познакомить учеников с написанием программ на языке, который будет изучаться дальше на уроках, что позволит им легче воспринимать материал по программированию в основной школе и развивать свои способности в этом направлении по индивидуальной образовательной траектории — как на уроках, так и во внеурочной деятельности.

Ключевые слова: информатика, начала программирования, пропедевтика программирования, программирование графических объектов, структурное программирование, язык PascalABC, язык Python, язык C++ для начинающих, КуМир, Рисователь.

* Материалы к статье можно скачать на сайте ИНФО: http://infojournal.ru/journals/school/school_06-2022/

Контактная информация

Гербут Светлана Сергеевна, учитель информатики, лицей № 2, г. Тула, Россия; адрес: 300002, Россия, г. Тула, ул. Галкина, д. 14; e-mail: gerbut-ss@yandex.ru

S. S. Gerbut

Lyceum 2, Tula, Russia

PROPAEDEUTICS OF PROGRAMMING: HOW TO PREPARE STUDENTS FOR THE PERCEPTION OF OBJECT-ORIENTED PROGRAMMING

Abstract

The article describes the methodological foundations for preparing for programming in basic school and, in particular, for the perception of object-oriented programming in the future. This methodics can be implemented in the course of extracurricular activities or use its fragments for implementation in the lesson. Programming is one of the most important sections in the basic school informatics course. This is an interesting and at the same time quite difficult section for initial learning. Given that the basic informatics course starts in the seventh grade and is only one hour per week (advanced — two hours per week), programming is allocated only in the eighth grade and not so much. And you need to introduce pupils to a large number of concepts, teach them how to write programs, from linear programs to array processing. In this regard, it is proposed to use a propaedeutic course of extracurricular activity for students of the sixth-seventh grades in order to familiarize them with the basics of structured programming using the example of working with performers, which allow them to realize familiarity with the structure of the program, syntactic requirements, basic algorithmic constructions through drawing and movement of graphic objects (according the teacher's choice — in algorithmic language, PascalABC, Python, C++). The methodological developments described in the article can be used for those performers who will not be in the lessons, or, conversely, to introduce students to writing programs in a language that will be studied further in the lessons, which will make it easier for them to perceive the material on programming in basic school and develop their abilities in this direction along an individual educational trajectory — both in the classroom and in extracurricular activities.

Keywords: informatics, basics of programming, propaedeutics of programming, programming of graphic objects, structural programming, PascalABC, Python, C++ for beginners, KuMir, Drawman.

1. Введение

В настоящее время все большей популярностью пользуется такое направление в информатике, и в частности в программировании, как раннее знакомство учащихся с объектно-ориентированным программированием (ООП). ООП можно рассматривать как модульное программирование нового уровня, когда вместо во многом случайного, механического объединения процедур и данных главным становится их смысловая связь [3].

Объектно-ориентированное программирование предполагает такой взгляд на моделируемые объекты, при котором они рассматриваются как единое целое со всем многообразием свойств и особенностей поведения [5].

Многие учителя являются сторонниками перехода к понятию объектов, методов и других понятий ООП только после изучения структурного программирования через консольные приложения. Чтобы все успеть рассказать и показать ученикам, конечно, надо ответить на острый вопрос: когда начать? Ведь обучение программированию в основной школе начинается, как правило, во второй половине восьмого класса, что довольно поздно в том смысле, что алгоритмическое мышление у них уже в основной массе сформировано. Понятие алгоритмического мышления тесно связано с понятием логического мышления. Психологи утверждают, что для развития логического мышления природой отведены определенные возрастные рамки, примерно соответствующие по срокам обучению в начальной школе.

Описанные ниже методические аспекты и структура пропедевтического курса нацелены на раннее знакомство с «настоящим» программированием в среде, в которой можно программировать серьезные проекты. Знакомство с программированием графических элементов, анимации в шестом-седьмом классах и с началами программирования в восьмом классе позволит ученикам восьмых-девярых классов уже выбрать направление программирования, которое им будет интересно для дальнейшего саморазвития (олимпиадное программирование, проектное программирование (ООП), работа с базами данных, веб-программирование и др.) и, возможно, для реализации индивидуального учебного проекта в старшей школе.

В данной статье подробно описано итоговое занятие по программированию, на котором учащиеся должны продемонстрировать полученные знания, проявить свои творческие и лидерские качества, потренироваться работать в команде. Такое занятие необходимо учителю для оценивания результатов своей работы по подготовке детей к восприятию ООП. Оценка — это всегда измерение определенных характеристик достигнутого образовательного результата. Объектом педагогической оценки в учебном процессе является деятельность обучающихся, а предметом — оценивание результатов этой деятельности, выраженное в различных качественных характеристиках и количественных показателях [7].

Для понимания значения этого занятия опишем сначала общую структуру пропедевтического курса.

2. Выбор среды программирования

Программирование графики (рисование объектов с помощью графических примитивов, анимация) —

это раздел программирования, с которого легче начать знакомство учеников с программированием и который наиболее интересен для учащихся.

Выбор конкретного языка программирования и среды программирования во многом зависит от используемой в конкретной школе программы обучения и от навыков учителя:

- для знакомства с программированием в шестом классе можно начать со среды «КуМир», выбрать исполнителя Рисователь;
- если в основной школе изучается язык программирования PascalABC, то можно и в пропедевтическом курсе сразу начать с него (с использованием модуля GraphABC);
- для знакомства с современным языком программирования Python можно воспользоваться авторской библиотекой К. Ю. Полякова graph.py;
- если цель учителя — в дальнейшем перейти к олимпиадному программированию, то можно выбрать язык C++ (в частности, начать работу с библиотеки TXXLib).

Если требуется написать программу с графическими возможностями на языке Python, обычно используют библиотеки, которые скрывают от программиста сложности взаимодействия с операционной системой. Одна из библиотек, позволяющая работать с графикой на языке Python, — это tkinter (устанавливается вместе с интерпретатором Python). Библиотека graph, с которой намного проще начать знакомство с программированием графики, является «оберткой» библиотеки tkinter [8].

Писать простые графические программы на C++ не всегда просто. Для вывода графики требуется с помощью операционной системы создать специальное окно, обеспечить его взаимодействие с пользователем и т. п. Для того чтобы упростить вход в мир графического программирования, были разработаны специальные библиотеки. Наиболее удобна библиотека TX Library, разработанная И. Р. Дединским (см. [8]).

Модуль GraphABC в PascalABC представляет собой простую графическую библиотеку и предназначен для создания несобытийных графических и анимационных программ в процедурном и, частично, в объектном стиле. Рисование осуществляется в специальном графическом окне, возможность рисования в нескольких окнах отсутствует. Кроме того, в модуле GraphABC определены простейшие события мыши и клавиатуры, позволяющие создавать элементарные событийные приложения [9].

PascalABC.NET — мультипарадигменный язык. Он позволяет программировать в структурном, объектно-программированном и функциональном стилях, а также смешивать эти стили в одной программе. Большое количество «синтаксического сахара» делает программирование на PascalABC.NET простым и комфортным как для начинающих, так и для имеющих опыт в создании программ [6].

Что общего у перечисленных выше сред программирования? В них используется компьютерная система координат, имеется набор похожих графических примитивов, возможность реализации процедурного программирования, в PascalABC и C++ можно продолжить изучение в направлении разработки собственных



Рис. 1. Компьютерная система координат

классов с объектами и методами (осуществить плавный переход к понятиям ООП).

Компьютерная система координат (рис. 1) характеризуется следующими особенностями: ось x — горизонтальная, направлена вправо, ось y — вертикальная, направлена вниз, точка $(0, 0)$ — начало координат находится в верхнем левом углу, все координаты — неотрицательные целые числа.

3. Темы курса внеурочной деятельности по программированию графики

Предполагается, что до итогового занятия учащиеся уже прошли вводное обучение программированию графики в выбранной учителем среде по следующим темам:

- Знакомство с компьютерной системой координат.
- Рабочая область исполнителя, программирование рисования графических примитивов.
- Управление цветом фона, заливки, линий фигур (встроенные цвета).
- Добавление в программу цветовых констант (в системе RGB).
- Знакомство с переменными. Целый и вещественный тип. Тип для цвета.
- Пересчет координат рисования объектов через переменные.
- Рисование одинаковых фрагментов из разных точек и разными цветами.
- Знакомство с процедурами и функциями.
- Функции с параметрами (координаты, цвет).
- Вывод текста на область рисования. Анимация через циклы.
- Программирование фрагментов с повторениями.
- Проект «Многоэтажный дом» с разными по цвету окнами (выключен и включен свет, цвет занавесок).

4. Сценарий итогового занятия (командная работа)

Тема занятия: Разработка проекта «Программируем созвездия»

Цель занятия: создание условий для развития алгоритмического мышления и творческого самовыражения школьников через программирование проектов, способствующих их будущим развитию и самосовершенствованию в области объектно-ориентированного программирования.

Задачи занятия:

- закрепить понятие алгоритма, структурного программирования путем деления алгоритма на под-

задачи и реализации в виде функций; подвести обучающихся к понятиям объектно-ориентированного программирования: объект, свойство, метод;

- сформировать у учащихся навыки работы в команде: распределить обязанности в команде по выполнению фрагментов задания, выбрать ответственного за объединение работ в единый проект; развивать творческие способности каждого школьника при реализации им выделенной части проекта;
- воспитывать у учащихся умения: работать в команде, высказывать свою точку зрения, прислушиваться к чужому мнению, находить оптимальное решение, принимать мнение команды, оказывать помощь напарнику и, наоборот, просить о помощи; способствовать передаче опыта от старшеклассников к более младшим ученикам.

Целевая аудитория: ученики шестого-седьмого классов.

Характеристика класса (группы).

Учащиеся:

- хорошо владеют навыками работы в выбранной среде программирования;
- понимают основы работы с компьютерной системой координат;
- знают структуру программы;
- владеют навыками создания линейных программ;
- умеют добавлять собственные цвета в программу;
- умеют писать программы с использованием переменных;
- понимают назначение подпрограмм;
- могут реализовать подпрограмму с параметрами (как минимум с тремя: координаты, цвет).

На некоторых занятиях уже использовалась работа в парах, но опыта работы в команде более двух человек у учеников не было.

Подготовительный этап.

В зависимости от уровня освоения материала учащимися к текущему занятию учителю необходимо спланировать задание (внести какие-то изменения — упростить или усложнить отдельные фрагменты).

Вопросы и ответы на них для этапа «Актуализация опорных знаний» следует оформить в слайдах презентации, чтобы можно было подкрепить ответы учеников фактическими командами, напомнить изученный материал. Наличие презентации на занятии обязательно, так как в активном восприятии и осмыслении изучаемого материала большое значение имеет умение учителя придавать изложению увлекательный характер, делать его живым и интересным [1].

Учитель должен продумать, как (в команды по сколько человек) можно разделить группу, посещающую внеурочные занятия, чтобы получилось несколько команд-участников. При небольшой посещаемости это может быть и одна команда. Если команд несколько, стоит пригласить старшеклассников для курирования работы каждой команды, обсудить с этими ребятами их роль в проведении занятия.



Рис. 2. Изображения созвездий (<https://mychatte.ru/catalog/raznoe/nabor-zodiakalnykh-sozvezdiy-/>)

Для выполнения задания надо подготовить изображения созвездий (один из примеров приведен на рисунке 2). Можно подготовить и любые другие изображения, главное — чтобы их можно было распечатать на белом фоне. Ученики с помощью линейки и карандаша должны будут разлиновать рисунок, чтобы просчитать координаты для рисования звезд выбранного созвездия в компьютерной системе координат.

Надо заранее распечатать карточки, каждая из которых содержит:

- формулировку задания;
- рисунки созвездий;
- таблицу «Командная работа» (табл. 1) в помощь ученикам для организации командной работы.

Также следует подготовить распечатки-подсказки по пройденным темам, которые могут понадобиться слабым ученикам во время выполнения задания.

Необходимо обратить особое внимание на подготовку техники: у каждого ученика в команде должно быть рабочее место (компьютер или ноутбук с установленным ПО); места должны быть распределены так, чтобы участники одной команды могли общаться друг с другом и видеть работу друг друга.

Учитель должен заранее подготовить поощрения за участие, это могут быть грамоты победителям, сертификаты участникам, памятные призы и т. п. — на усмотрение учителя.

Занятие рассчитано на 2,5 часа (три академических урока).

Таблица 1

Командная работа

№ п/п	Работа в команде	Отметка о выполнении
1	Выбрать ученика, который будет готовить диалог программы и собирать проект в единое целое	
2	Обсудить, на какие блоки разбивается алгоритм	
3	Обсудить реализацию диалога с пользователем	
4	Обсудить шаблон подпрограмм, названия, параметры	
5	Распределить созвездия между участниками	
6	Договориться о размере окна рисования	
7	Договориться о размерах и формах звезд	
8	Обсудить цветовую гамму проекта	
9	Обсудить наличие или отсутствие анимации	
10	На распечатках созвездий просчитать координаты каждой звезды	

Таблица 2

Ход занятия

№ этапа	Этап занятия	Время, мин	Деятельность учителя	Деятельность учащихся
1	Организационный момент	3	Приветственное слово. Озвучивает цель занятия: реализовать командный проект «Программируем созвездия» с применением полученных знаний в программировании. Поясняет, что основной формой деятельности на занятии будет командная работа. Представляет старшеклассников, поясняет, что их роль не реализовать проект, а помочь, подсказать в организационных вопросах, ответить на вопросы по программе.	Приветствуют учителя, старшеклассников. Задают организационные вопросы.

Продолжение табл. 2

№ этапа	Этап занятия	Время, мин	Деятельность учителя	Деятельность учащихся
2	Актуализация опорных знаний	15	Предлагает ученикам вспомнить основные особенности программирования в изучаемой среде: 1) Что такое «компьютерная система координат»? 2) Какова структура программы на выбранном языке? 3) Какие вам известны команды исполнителя для рисования графических примитивов? 4) Какие вам известны команды для работы с цветом? 5) Что такое переменные? Зачем они нужны? К какому типу данных они относятся? 6) Каким образом можно организовать рисование одинаковых фрагментов из разных точек и разными цветами? 7) Что такое анимация и какими средствами программирования ее можно реализовать?	Ответы учеников будут зависеть от используемой среды программирования и исполнителя. Но общая идея одинакова: 1) Учащиеся должны вспомнить, что начало координат находится в верхнем левом углу рабочей области. 2) Структура программы зависит от конкретного языка программирования. 3) Линия, прямоугольник, окружность, эллипс и др. (конкретные названия команд зависят от используемого языка программирования). 4) Цвет линии, цвет заливки, задание цветов в модели RGB. 5) Нужны для упрощения рисования похожих объектов из разных точек. Есть целые и вещественные переменные. 6) С помощью подпрограмм, подпрограмм с параметрами. 7) Учащиеся должны вспомнить известные циклы (количество итераций, с параметром, с предусловием — ответы зависят от того, что учитель рассказывал ранее).
3	Деление на команды	3	Руководит распределением учащихся по командам.	Учащиеся делятся на команды — команду набирает капитан или распределение по командам определяется жеребьевкой.
4	Формулировка задания	5	Объясняет задание : Разработать проект, в котором будет реализована демонстрация некоторых созвездий (из распечаток). В программе должны быть реализованы диалог между пользователем и программой, позволяющий выбрать созвездие для демонстрации, а также возможность очистить область рисования. Поскольку это командная работа, следует проговорить необходимость распределения обязанностей между участниками в команде.	Слушают задание, задают уточняющие вопросы.
5	Распределение учащихся по рабочим местам	2	Размещает учеников так, чтобы они могли общаться с остальными членами команды.	Сдвигают парты, чтобы каждая команда работала за одним большим столом, или просто занимают заранее подготовленные учителем места.
6	Работа в команде	17	Координирует командную работу, при этом старшеклассники помогают ему следить за распределением членов каждой команды по пунктам, указанным в карточке с заданием.	Обсуждают распределение обязанностей по программированию отдельных блоков программы. Выбирают члена команды, который будет собирать проект в единое целое. Обсуждают, на какие подпрограммы будет разбиваться алгоритм, наличие параметров в функциях. Распределяют между участниками фрагменты задания. Договариваются о размерах окна рисования, о цветовой гамме визуализации проекта, о размерах звезд, наличии или отсутствии анимации. На распечатках созвездий просчитывают координаты каждой звезды.

Окончание табл. 2

№ этапа	Этап занятия	Время, мин	Деятельность учителя	Деятельность учащихся
	Перерыв	10		
7	Индивидуальная работа каждого участника	45	Вместе со старшеклассниками следит за работой команд, отвечает на вопросы. Каждые 15 минут напоминает учащимся о необходимости проведения гимнастики для глаз.	Ученики за компьютерами выполняют свою часть проекта, они могут обращаться за помощью или советом к участникам своей команды, к старшекласснику.
			У учителя заранее подготовлены фрагменты программ с предыдущих уроков на случай, если кто-то из учеников ошибается в синтаксисе или структуре программы. Но использование таких подсказок учитывается при подведении итогов. У учителя должны быть подготовлены распечатки с созвездиями для раздачи учащимся в случае дополнительного запроса.	Капитан каждой команды создает заготовку проекта, в которую будут добавлены все остальные части. В случае быстрого выполнения какими-то участниками своей части работы команда может принять решение добавить в программу анимацию или дополнительные созвездия.
	Перерыв	10		
8	Работа в команде	20	Вместе со старшеклассниками следит за правильностью сохранения программы на флеш-накопитель или в сетевую папку, передачи кода капитану команды, помогает при возникновении трудностей.	Ответственный член команды (обычно капитан) объединяет полученные от участников команды решения в общий проект. Команда обсуждает выступление на защите.
9	Демонстрация проектов командами, защита	20	Учитель и старшеклассники: • слушают выступления команд; • задают вопросы; • оценивают проекты по критериям (табл. 3); • награждают участников грамотами и памятными призами (если это предусмотрено).	Демонстрируют проекты. Выступают перед другими командами: рассказывают о том, что сделано в работе, что не получилось реализовать, какие варианты развития проекта они видят, отвечают на вопросы.
10	Подведение итогов, рефлексия	5	Благодарит всех участников за их работу. Отмечает, что сегодня они программировали собственные объекты, у которых есть свойства: цвет, местоположение и др., в каких-то проектах появились методы: находиться на месте, двигаться. Это задел на будущее объектно-ориентированное программирование. Подводит всех к мысли, что программирование имеет огромные масштабы для реализации творческих замыслов, что нужно его учить более углубленно и создавать свои собственные разработки. Отмечает помощь старшеклассников. Подчеркивает, что работа в команде — это непростое, но продуктивное дело. Все молодцы, что смогли распределить задания, помочь друг другу справиться с трудностями и в итоге представить отличный совместный проект.	Высказываются о проделанной работе, обсуждают, в чем разница между индивидуальной и командной работой, с какими трудностями они столкнулись, какие проекты им бы хотелось еще реализовать.

Примеры проектов, выполненных учащимися, а также задание и презентацию к занятию можно скачать на сайте ИНФО в дополнительных материалах к статье:

http://infojournal.ru/journals/school/school_06-2022/

Оценивание.

Каждая работа оценивается по определенным критериям (табл. 3), оценка по каждому критерию — до 3 баллов, при этом:

- 0 — полное отсутствие (критерий не выполняется);
- 1 — частичная реализация;
- 2 — реализация хорошая, но есть неточности;
- 3 — отличная реализация.

5. Заключение

В рамках описанного выше занятия у учащихся в разной степени формируются все четыре вида универсальных учебных действий: личностные, регулятивные, познавательные, коммуникативные. Они обеспечивают возможность учащегося самостоятельно осуществлять деятельность учения, ставить учебные цели, искать и использовать необходимые средства и способы достижения этих целей, контролировать и оценивать процесс и результаты своей и коллективной деятельности [2].

Подготовленный сценарий занятия можно использовать как:

- конкурсное (состязательное) мероприятие;

Таблица 3

Критерии оценки работ

№ п/п	Критерии	Баллы
1	Объединение программ в единый проект	
2	Реализация диалога программы с пользователем	
3	Структура программы в одном шаблоне (название подпрограмм и порядок параметров при наличии)	
4	Написание подпрограмм с параметрами	
5	Единый стиль оформления результата (цвет, форма и размер звезд)	
6	Наличие анимации	
7	Просчет координат на распечатках	
	Штрафной балл за использование подсказок учителя (вычитается из общей суммы)	
	ИТОГО:	

- зачетную работу;
- занятие для подготовки команды к участию в сто-
роннем конкурсе.

Описанный выше сценарий можно доработать до самостоятельного мероприятия, целью которого будет знакомство со структурным программированием (для новичков) для привлечения внимания школьников к программированию и, возможно, с целью дальнейшего посещения курса внеурочной деятельности. Данное мероприятие будет рассчитано уже на целый день (т. е. полностью меняется формат его проведения).

Подготовка к объектно-ориентированному программированию начинается со структурного программирования, а именно с умения написать процедуры и функции — разного назначения, но имеющие что-то общее. Тогда ученики смогут разобраться с понятиями «объект», «свойство объекта», «метод». На примере вышеописанного занятия объектами являются звезды, у которых есть свойства: цвет, местоположение (координаты), размер и др., а также методы: рисование звезды, движение звезды. Это самое наглядное, но практика показывает, что кто-то из учеников обязательно заметит, что также объектом являются область рисования, диалог с программой и т. д. Это большой задел на будущее изучение объектно-ориентированного программирования.

Список источников

1. Босова Л. Л., Босова А. Ю. Информатика: методическое пособие для 5–6 классов. М.: БИНОМ. Лаборатория знаний, 2014. 384 с.
2. Босова Л. Л., Босова А. Ю. Информатика: методическое пособие для 7–9 классов. М.: БИНОМ. Лаборатория знаний, 2015. 472 с.
3. Гвоздева В. А. Введение в специальность программиста: учебник. 2-е изд., испр. и доп. М.: ИД «ФОРУМ»; ИНФРА-М, 2007. 208 с.
4. Голицына О. Л., Попов И. И. Программирование на языках высокого уровня: учебное пособие. М.: ФОРУМ, 2008. 496 с.
5. Куперман В. Г., Торина Е. Г. Основы информатики и начала программирования: учебное пособие. Тула: Изд-во Тульского государственного педагогического университета им. Л. Н. Толстого, 1997. 264 с.
6. Осипов А. В. PascalABC.NET: Введение в современное программирование. Ростов-на-Дону, 2019. 572 с.
7. Основы общей теории и методики обучения информатике: учебное пособие / под ред. А. А. Кузнецова. М.: БИНОМ. Лаборатория знаний, 2010. 207 с.
8. Поляков К. Ю. Программирование. Python. C++. Ч. 1: учебное пособие. 2-е изд., испр. М.: БИНОМ. Лаборатория знаний, 2020. 144 с.
9. Система программирования PascalABC.NET. <http://pascalabc.net/>