



УДК 378.147:004.92**РАЗРАБОТКА И АПРОБАЦИЯ УЧЕБНОГО САЙТА ПО ГРАФИКЕ В
VISUAL C#****Градинар Игорь Игоревич,**магистрант, Кубанский государственный университет, г. Краснодар, Российская
Федерация. E-mail: gradinar@kubsu.ru.**Царева Ирина Николаевна,**кандидат педагогических наук, доцент, Кубанский государственный университет, г.
Краснодар, Российская Федерация. E-mail: tsareva@kubsu.ru**Аннотация**

В статье описаны проектирование и пилотная апробация учебного материала по графике в Visual C# с использованием Windows Forms и GDI+. Изложены дидактические принципы, на которых строилось содержание: нарастающая сложность, модульность и критериальное оценивание. Представлена модульная архитектура из восьми тематических блоков (обязательная часть – 45–55 часов), реализованная как статический веб-сайт с системой контроля прогресса на основе локального хранилища. В апробации приняли участие 24 учащихся; 18 из них (75 %) завершили обязательную часть, средний балл итогового теста составил 74 %. Выявлены модули, вызвавшие наибольшие затруднения, и показано, как корректировка подачи материала повлияла на результаты. Рассмотрены методические ограничения исследования (малая выборка, совмещение автором ролей разработчика и преподавателя) и намечены направления дальнейшей работы.

Ключевые слова: C#, GDI+, Windows Forms, графическое программирование, контроль прогресса обучаемого, пилотная апробация, учебный материал

**DEVELOPMENT AND PILOT TESTING OF AN EDUCATIONAL WEBSITE
ON GRAPHICS IN VISUAL C#****Gradinar Igor Igorevich,**Master's student, Kuban State University, Krasnodar, Russian Federation. E-
mail: gradinar@kubsu.ru.**Tsareva Irina Nikolaevna,**Candidate of Pedagogical Sciences, Associate Professor, Kuban State University, Krasnodar,
Russian Federation. E-mail: tsareva@kubsu.ru

ABSTRACT

The article describes the design and pilot testing of educational material on graphics in Visual C# using Windows Forms and GDI+. The didactic principles underlying the content – increasing complexity, modularity and criterion-based assessment – are outlined. The modular architecture consists of eight thematic units (compulsory part – 45–55 hours) and is implemented as a static website with a progress control system based on local storage. Twenty-four students participated in the pilot; 18 of them (75 %) completed the compulsory part, with an average final test score of 74 %. The modules that caused the greatest difficulties are identified, and it is shown how adjustments to the presentation affected the results. Methodological limitations of the study (small sample size, the author acting both as developer and instructor) are discussed, and directions for further work are outlined.

Keywords: C#, GDI+, Windows Forms, graphical programming, learner progress control, pilot testing, educational material.

Актуальность. Графическое программирование занимает заметное место в подготовке разработчиков программного обеспечения: визуальный интерфейс в том или ином виде присутствует практически в любом современном приложении, от элементов управления до диаграмм и картографических компонентов. У этой темы есть и дидактическое преимущество: результат выполнения кода виден сразу, что обеспечивает быструю обратную связь и поддерживает учебную мотивацию при освоении программирования [8, 9].

В русскоязычной учебной литературе графическое программирование освещается фрагментарно. Имеющиеся учебные материалы по Visual C# и .NET чаще ориентированы на синтаксис языка, среду разработки и общую архитектуру платформы, тогда как систематизированное изложение средств графического вывода в Windows Forms встречается значительно реже [3]. На крупнейших отечественных учебных площадках курсы по C# носят общий характер: вопросы графического вывода в них либо не выделены в самостоятельную траекторию, либо представлены отдельными примерами [4, 6]. Этот дефицит и определил цель работы: разработать, реализовать и провести пилотную апробацию методического пособия по графическому программированию на C#/Windows Forms/GDI+, обеспечивающего последовательное и практикоориентированное освоение темы.

За рубежом обучение компьютерной графике в привязке к языку программирования часто строится через интерактивные среды с немедленной визуальной обратной связью: Processing, p5.js, Pygame [8, 9]. В исследованиях и учебных изданиях по creative coding визуальный проект рассматривается не как иллюстрация после изучения темы, а как основная форма закрепления программных понятий: студент получает завершённый видимый результат и может изменять его через код [8, 9].

На платформе .NET ситуация иная: систематических открытых учебных ресурсов по GDI+ в русскоязычном сегменте немного. Каталог Stepik и программа Яндекс Практикума ориентированы преимущественно на общий курс C#, а справочный раздел Microsoft Learn описывает API, но не заменяет последовательную учебную траекторию по графическому выводу [4, 5, 6]. Это подтверждает методический дефицит и обосновывает разработку отдельного ресурса.

Идеи, заложенные в исследованиях адаптивных учебных траекторий [7], использованы при проектировании системы управления прогрессом в разработанном материале.

Цель исследования. Разработать и провести пилотную апробацию учебного материала по графическому программированию на Visual C#/Windows Forms/GDI+ для студентов 1-2 курсов, обеспечивающего последовательное и практикоориентированное освоение темы на основе принципов нарастающей сложности, модульности и критериального оценивания.

Материалы и методы исследования. Учебный материал рассчитан на учащихся, умеющих разрабатывать примитивные приложения в Visual C#, освоивших базовый синтаксис C# и событийную модель .NET: без понимания механизма событий освоение графического конвейера Windows Forms существенно затрудняется.

Планируемые результаты обучения сформулированы в терминах операциональных компетенций (по таксономии Блума — уровни «применение» и «анализ»). По завершении пособия учащийся должен уметь:

- понимать архитектуру графического вывода в Windows Forms и назначение события Paint;

- применять классы Graphics, Pen, Brush и базовые примитивы для построения двумерных сцен;

- реализовывать анимацию с использованием двойной буферизации;

- обрабатывать события мыши и клавиатуры для создания интерактивных приложений;

- строить графики и диаграммы с корректным пересчётом математических координат в экранные;

- применять базовые алгоритмы обработки изображений (фильтры, свёртка).

Учебный материал выстроен по нарастающей сложности: освоение архитектуры графического окна предшествует инструментам рисования, а оптимизация и интерактивность вводятся только после того, как закреплены базовые навыки. Это соответствует логике зоны ближайшего развития [1]. При этом часть тематических блоков остаётся достаточно автономной — например, модуль по построению графиков и диаграмм можно при необходимости проходить вне общей последовательности, если этого требует учебный план.

Внутри каждого урока сохраняется одна и та же трёхчастная структура: теория, практика, контроль; в дидактике такая последовательность рассматривается как устойчивая модель формирования учебного действия [2]. Абстрактных упражнений «для тренировки» учебный материал не предусматривает: каждое задание представляет собой фрагмент реальной задачи, который студент должен запустить и изменить самостоятельно, без участия преподавателя.

Переход к следующему уроку открывается только при результате теста не ниже 70%. Порог установлен на уровне, который в принятой модели критериального оценивания соответствует уверенному владению базовыми понятиями темы: более низкий результат, по наблюдениям в рамках пособия, обычно означает накопление пробелов, которые затрудняют освоение следующего модуля.

В отличие от типовых пособий по C#, где графика либо отсутствует, либо дана как набор изолированных примеров, в этом пособии каждый блок подчинён общей логике, а не существует сам по себе.

Учебный материал состоит из восьми модулей (табл. 1). Модули 1-6 составляют обязательную часть объёмом 45-55 аудиторных часов; модули 7-8 вынесены в дополнительный блок, который преподаватель включает по своему усмотрению.

Таблица 1 – Модульная структура пособия

Модуль	Основное содержание	Объём, ч.
Первые шаги в графике	Архитектура Windows Forms, система координат, событие Paint	6–8
Инструменты рисования	Graphics, Pen, Brush, примитивы, цвет, текст	10–12
Оптимизация и буферизация	Двойная буферизация, Invalidate, LockBits	5–6
Анимация и интерактивность	События мыши и клавиатуры, таймеры, трансформации	8–10
Построение графиков и диаграмм	Пересчёт координат, динамические данные	6–8
Продвинутые техники	Фильтры изображений, фракталы, текстуры	8–10
Современные технологии	Введение в WPF, сравнение с GDI+	4–6
Анимации и эффекты	Системы частиц, 3D-куб, физика столкновений	10–12

Источник: исследование проведено авторами статьи.

Первый модуль «Первые шаги в графике» содержит информацию о системе координат GDI+: главная сложность — направление оси ординат. В GDI+ она направлена вниз, что противоречит привычной для студентов математической записи. В рамках апробации (раздел 7) именно инверсия оси Y оказалась причиной ошибок у 9 из 24 студентов (37,5%) в первом практическом задании модуля 5. После того как в модули 1 и 5 добавили визуальные схемы соответствия математической и экранной систем координат, доля таких ошибок в следующей группе заметно снизилась — до уровня единичных случаев. Этот результат нуждается в проверке на большей выборке, прежде чем делать обобщённые выводы. В модуле 3 («Оптимизация и буферизация») производительность пиксельных операций демонстрируется через сравнение методов доступа к изображению: GetPixel показывает поэлементный доступ, LockBits — работу с данными изображения через блок памяти [5]. В модуле 4 («Анимация и интерактивность») студенты разрабатывают приложение с движущимся объектом, управляемым таймером, и реализуют его перемещение с помощью клавиатуры или мыши, что позволяет закрепить навыки обработки событий и работы с трансформациями. В модуле 5 («Построение графиков и диаграмм») основное задание — визуализация математической функции, например построение синусоиды, вписанной в клиентскую область формы. В этом задании от студента требуется пересчёт математических координат в экранные с учётом инвертированной оси Y — именно здесь, как показано выше, возникали основные ошибки на апробации. В остальных модулях аналогично описываются теоретические и практические примеры.

Учебный материал реализован как статический веб-сайт, не требующий серверного программного обеспечения: он одинаково работает и с локального диска, и при размещении на обычном веб-хостинге. Стек минимален — HTML5, CSS3 и JavaScript без фреймворков; для подсветки синтаксиса C# используется библиотека Prism.js. Отказ от фреймворков — осознанное решение в пользу долгосрочной сопровождаемости и минимальной зависимости от внешних экосистем, период актуальности которых для учебного ресурса зачастую ограничен.

Учебный ресурс включает 53 страницы, из которых 42 — учебные уроки. Каждый урок состоит из теоретического блока с примерами кода, пошагового практического

задания и теста из 5–10 вопросов. Прогресс студента сохраняется в localStorage браузера; за переходы между уроками отвечает модуль ProgressManager, открывающий доступ к следующему уроку только после прохождения теста текущего с результатом не ниже 70%. Это техническое решение — прямое продолжение принципа критериального оценивания из раздела 4, а не отдельная инженерная находка.

Результаты и их обсуждение. Апробация разработанного учебного материала проводилась в весеннем семестре 2025/26 учебного года в рамках дополнительного курса по программированию в онлайн-школе программирования ischool, который вёл автор. В ней участвовали 24 студента 1–2 курсов, имеющих предварительную подготовку по объектно-ориентированному программированию на C#. Учебный материал использовался в качестве основного учебного ресурса; занятия проводились дистанционно, на личных устройствах учащихся, а самостоятельная работа выполнялась в индивидуальном формате.

Совмещение автором роли разработчика и преподавателя в апробации — заметное методическое ограничение: оно не позволяет исключить влияние личной заинтересованности и контакта «учитель — ученик» на итоговые показатели. Это ограничение будет рассмотрено ниже.

Результаты оценивались тремя способами: журналом автоматической регистрации результатов прохождения тестов, с использованием ProgressManager; итоговым контрольным тестом из 30 вопросов по всем шести обязательным модулям; и анкетой субъективной оценки (шкала Лайкерта, 1–5, 12 утверждений по четырём конструктам — доступность изложения, прикладная направленность, качество заданий, общая удовлетворённость), которую ученики заполняли после завершения работы с учебным материалом.

О курсе

Этот курс научит вас создавать красивые графические приложения на C# с использованием GDI+ и Windows Forms. Вы пройдёте путь от создания первого окна до реализации сложных визуализаций и спецэффектов.

8	42	50-60	8+
Модулей	Уроков	Часов	Проектов

Модули курса

Модуль 1 Первые шаги в графике 5 уроков

Создаем первое окно и учимся в нем рисовать

- Основы C# и .NET
- Основы графического программирования
- От Консоли к Окну
- Жизненный цикл рисования
- Расположение элементов управления

[Начать модуль →](#)

Модуль 2 Учимся рисовать фигуры 8 уроков

Линии, круги, прямоугольники и работа с цветом

- Введение в GDI+

Рисунок 1 – Главная страница учебного сайта

Рисунок 1 демонстрирует главную страницу учебного сайта, на которой представлен общий интерфейс ресурса и доступны основные разделы курса.

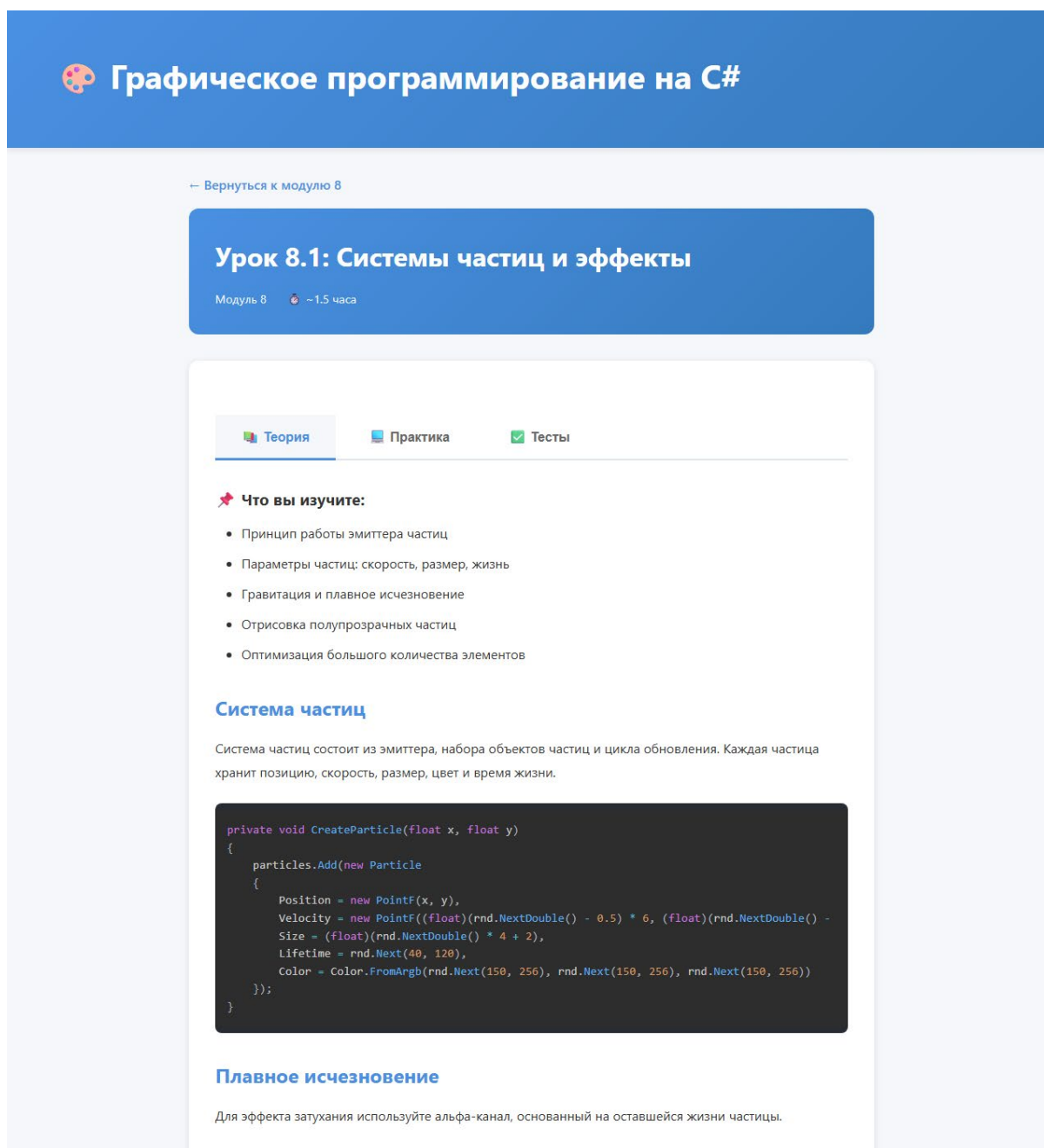


Рисунок 2 – Страница урока с теоретическим материалом

Рисунок 2 показывает страницу урока с теоретическим материалом, где изложены основные понятия и объяснения темы.

The screenshot displays a web-based learning interface for 'Графическое программирование на C#' (Graphics programming in C#). At the top, there's a blue header with the course title. Below it, a navigation bar includes a link to 'Вернуться к модулю 8' (Return to module 8). The main content area is titled 'Урок 8.1: Системы частиц и эффекты' (Lesson 8.1: Particle systems and effects) and indicates 'Модуль 8' (Module 8) with a duration of '1.5 часа' (1.5 hours). A tabbed interface shows 'Теория' (Theory), 'Практика' (Practice), and 'Тесты' (Tests), with 'Практика' being the active tab. Under 'Практическое задание' (Practical task), there are four bullet points: 'Реализуйте эмиттер, который создаёт новые частицы каждый тик.' (Implement an emitter that creates new particles every tick.), 'Добавьте гравитацию и уменьшение скорости.' (Add gravity and speed reduction.), 'Рисуйте частицы с помощью FillEllipse и полупрозрачной кисти.' (Draw particles using FillEllipse and a semi-transparent brush.), and 'Очищайте устаревшие частицы.' (Clean up outdated particles.). Below this, a section titled 'Задание по аналогии: Система частиц' (Analogous task: Particle system) instructs to implement an emitter with physics, similar to Task21_ParticleAnimation. It lists five steps: 1. Create an internal Particle class with fields: Position, Velocity, Color, Size, Lifetime, MaxLifetime, Gravity. Add a calculated property Alpha = (Lifetime / MaxLifetime) * 255. 2. In the Particle.Update() method, update position by velocity (pos.X += vel.X), add gravity (vel.Y += Gravity), and decrease Lifetime by 1. 3. In the Timer_Tick method, create 3 new particles per tick (if < 500), then update all particles and remove dead ones from the end of the list. 4. In the Form_Paint method, for each particle, create a SolidBrush with Color.FromArgb(alpha, p.Color) and draw an ellipse with size p.Size. 5. Additionally, add several modes via ComboBox — fireworks.

Рисунок 3 – Пример практического задания с кодом

Рисунок 3 иллюстрирует пример практического задания с кодом, отражающий способ выполнения упражнения и применение изученного материала.

Из 24 учеников, приступивших к работе с учебным материалом, 18 (75%) завершили обязательную часть в установленные сроки. Трое прошли учебный материал частично (модули 1–4), ещё трое не приступили к самостоятельной работе — по причинам, не связанным с содержанием пособия (личная загруженность, изменение учебных планов).

Средний балл итогового теста по выборке учащихся завершивших учебный материал — 74% ($\sigma = 9,3\%$, медиана — 76%); распределение результатов близко к нормальному: трое набрали менее 60%, одиннадцать — от 70% до 84%, четверо — 85% и выше. При выборке такого объёма ($n = 18$) эти оценки следует воспринимать как ориентировочные, а не как статистически устойчивые показатели.

Таблица 2. – Средние баллы промежуточных тестов по модулям ($n = 18$)

Модуль	Средний балл, %	σ , %
Первые шаги в графике	81	7,1
Инструменты рисования	78	8,4
Оптимизация и буферизация	69	11,2

Анимация и интерактивность	75	9,8
Построение графиков и диаграмм	71	10,5
Продвинутое техники	66	12,3

Источник: исследование проведено авторами статьи.

Наиболее низкие результаты — в модулях 3 (оптимизация, 69%) и 6 (продвинутое техники, 66%); в модуле 6 заметно выше и разброс ($\sigma = 12,3\%$ против среднего по выборке 9,3%), что говорит о неравномерности усвоения. Модуль требует знакомства с основами обработки сигналов и матричными операциями, и вероятная причина разброса — разная исходная математическая подготовка студентов; прямой проверки этой гипотезы в рамках апробации не проводилось.

Анкету по пилотной апробации заполнили 17 из 18 завершивших учебный материал (94,4%). Средние оценки по конструктам: доступность изложения — 4,2 из 5; прикладная направленность заданий — 4,5 из 5; качество примеров кода — 4,3 из 5; общая удовлетворённость — 4,1 из 5. Минимальные оценки по доступности изложения получили те же модули 3 и 6, где результаты теста были самыми низкими (69% и 66% соответственно). Совпадение объективных и субъективных оценок указывает на то, что слабым местом, вероятно, является сама подача материала этих модулей, а не формат контроля — хотя без качественного анализа отзывов учеников это остаётся предположением.

Таблица 3. – Результаты анкетирования (n = 17)

№	Конструкт	Средняя оценка (1-5)
1	Доступность изложения	4,2
2	Прикладная направленность заданий	4,5
3	Качество примеров кода	4,3
4	Общая удовлетворённость	4,1

Источник: исследование проведено авторами статьи.

Главное ограничение апробации — её масштаб и условия проведения. Выборка (n = 24) слишком мала для статистически обоснованных выводов о педагогической эффективности пособия, а единственным преподавателем был сам автор пособия, что не позволяет отделить эффект материалов от эффекта личного взаимодействия с учениками. На следующий учебный год планируется апробация на двух группах (ориентировочно 48–50 студентов) с привлечением другого преподавателя хотя бы для части групп и, по возможности, с контрольной группой на традиционном учебнике.

Второе ограничение апробации — выбор технологии. GDI+ не является актуальным стандартом разработки GUI-приложений на .NET. Выбор сделан осознанно: GDI+ требует минимальной инфраструктурной подготовки, и студент видит результат своего кода на экране сразу после запуска, без дополнительной настройки окружения, которая неизбежна, например, при работе с XAML в WPF. Переход к WPF и MAUI вынесен в дополнительный модуль 7.

Третье ограничение апробации — техническое: localStorage как хранилище прогресса работает в пределах одного браузера и устройства. Для интеграции с системами управления обучением (LMS) потребуется переход на серверное хранение данных с поддержкой стандарта xAPI/Tin Can.

Выводы. Учебный материал по графике в Visual C#/Windows Forms/GDI+ был разработан и успешно прошёл пилотную апробацию. — 53 веб-страницы (42 урока), организованные в восемь тематических модулей с объёмом обязательной части 45–55 часов.

В основу проектирования были положены принципы нарастающей сложности, модульности и критериального оценивания, обязывающего пройти пороговый тест перед переходом к следующей теме.

По результатам пилотного внедрения ($n = 24$): 75% учащихся завершили обязательную часть учебного материала, средний балл итогового теста — 74% ($\sigma = 9,3\%$), анкетирование зафиксировало высокую оценку прикладной направленности заданий (4,5 из 5). Наибольшие затруднения выявлены в модулях 3 и 6; для модуля 6 намечена вероятная связь с неравномерной математической подготовкой студентов, для модуля 3 причины пока требуют отдельного анализа.

Результаты апробации носят предварительный характер прежде всего из-за малой выборки и совмещения автором ролей разработчика и преподавателя пособия. Их верификация на расширенной выборке, желательно с контрольной группой и независимым преподавателем, составит предмет последующей публикации.

Список литературы:

1. Зарецкий, В. К. Эвристический потенциал понятия «зона ближайшего развития» / В. К. Зарецкий // Вопросы психологии. – 2008. – № 6. – С. 13–25.
2. Хуторской, А. В. Дидактика как наука: предмет, задачи, методы / А. В. Хуторской // Вестник Института образования человека. – 2015. – № 2. – С. 1–12.
3. Проектирование графических интерфейсов в среде Visual Studio на базе C#: учебное пособие / В. В. Турупалов, Н. К. Андриевская, Т. В. Мартыненко, Е. А. Хрюкин; под общ. ред. В. В. Турупалова. – Москва; Вологда: Инфра-Инженерия, 2025. – 232 с.
4. Яндекс Практикум: программа обучения разработчика на C# [Электронный ресурс]. – URL: <https://practicum.yandex.ru/csharp-developer/> (дата обращения: 13.05.2026).
5. Microsoft Learn: GDI+ Managed Classes [Электронный ресурс]. – URL: <https://learn.microsoft.com/ru-ru/dotnet/desktop/winforms/advanced/using-managed-graphics-classes> (дата обращения: 18.05.2026).
6. Stepik: каталог курсов по C# [Электронный ресурс]. – URL: <https://stepik.org/catalog/search?query=C%23> (дата обращения: 18.05.2026).
7. Brusilovsky, P. Adaptive Hypermedia / P. Brusilovsky // User Modeling and User-Adapted Interaction. – 2001. – Vol. 11. – No. 1–2. – Pp. 87–110. DOI 10.1023/A:1011143116306.
8. Greenberg, I. Creative coding and visual portfolios for CS1 / I. Greenberg, D. Kumar, D. Xu // Proceedings of the 43rd ACM Technical Symposium on Computer Science Education (SIGCSE'12). – 2012. – Pp. 247–252. DOI 10.1145/2157136.2157214.
9. Shiffman, D. Learning Processing: A Beginner's Guide to Programming Images, Animation, and Interaction / D. Shiffman. – 2nd ed. – Burlington: Morgan Kaufmann, 2015. – 564 p.
10. Shoyqulov, Sh. Q. Graphical Programming of 2D Applications in C# / Sh. Q. Shoyqulov // Eurasian Journal of Mathematical Theory and Computer Sciences. – 2024. – Vol. 4. – No. 10. – Pp. 34–42.

References:

1. Zaretsky, V. K. Heuristic potential of the concept of “zone of proximal development” / V. K. Zaretsky // Voprosy Psikhologii. – 2008. – No. 6. – Pp. 13–25.

2. Khutorskoy, A. V. Didactics as a science: subject, tasks, methods / A. V. Khutorskoy // Bulletin of the Institute of Human Education. – 2015. – No. 2. – Pp. 1–12.
3. Designing graphical interfaces in Visual Studio based on C#: textbook / V. V. Turupalov, N. K. Andrievskaya, T. V. Martynenkova, E. A. Khryukin; ed. by V. V. Turupalov. – Moscow; Vologda: Infra-Engineering, 2025. – 232 p.
4. Yandex Practicum: C# developer training program. – URL: <https://practicum.yandex.ru/csharp-developer/> (accessed: 05/13/2026).
5. Microsoft Learn: GDI+ Managed Classes. – URL: <https://learn.microsoft.com/ru-ru/dotnet/desktop/winforms/advanced/using-managed-graphics-classes> (accessed: 05/18/2026).
6. Stepik: catalog of C# courses. – URL: <https://stepik.org/catalog/search?query=C%23> (accessed: 05/18/2026).
7. Brusilovsky, P. Adaptive Hypermedia / P. Brusilovsky // User Modeling and User-Adapted Interaction. – 2001. – Vol. 11. – No. 1–2. – Pp. 87–110. DOI 10.1023/A:1011143116306.
8. Greenberg, I. Creative coding and visual portfolios for CS1 / I. Greenberg, D. Kumar, D. Xu // Proceedings of the 43rd ACM Technical Symposium on Computer Science Education (SIGCSE'12). – 2012. – Pp. 247–252. DOI 10.1145/2157136.2157214.
9. Shiffman, D. Learning Processing: A Beginner's Guide to Programming Images, Animation, and Interaction / D. Shiffman. – 2nd ed. – Burlington: Morgan Kaufmann, 2015. – 564 p.
10. Shoyqulov, Sh. Q. Graphical Programming of 2D Applications in C# / Sh. Q. Shoyqulov // Eurasian Journal of Mathematical Theory and Computer Sciences. – 2024. – Vol. 4. – No. 10. – Pp. 34–42.

Приложение № 1

**АНКЕТА ПО ПИЛОТНОЙ АПРОБАЦИИ УЧЕБНОГО МАТЕРИАЛА ПО
ГРАФИКЕ В VISUAL C#**

Уважаемый участник! Просим вас оценить учебный материал по графике в Visual C#. Ваши ответы помогут улучшить качество пособия.

Инструкция. Оцените каждое утверждение по шкале от 1 до 5, где 1 — полностью не согласен, 5 — полностью согласен.

Шкала оценки: 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

Раздел 1. Доступность изложения материала

1.1. Теоретический материал изложен понятным языком.

Оценка: 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

1.2. Примеры кода сопровождаются подробными комментариями.

Оценка: 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

1.3. Переход от одной темы к другой осуществляется логично и последовательно.

Оценка: 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

1.4. Объяснений достаточно для самостоятельного выполнения заданий.

Оценка: 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

Раздел 2. Прикладная направленность заданий

2.1. Практические задания имеют реальную практическую ценность.

Оценка: 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

2.2. Полученные навыки применимы в реальных проектах разработки.

Оценка: 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

2.3. Примеры отражают типичные задачи графического программирования.

Оценка: 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

Раздел 3. Качество учебных материалов и заданий

3.1. Уровень сложности заданий соответствует подготовленности студента.

Оценка: 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

3.2. Инструкции к практическим заданиям сформулированы чётко и однозначно.

Оценка: 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

3.3. Тестовые вопросы адекватно проверяют усвоение материала урока.

Оценка: 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

Раздел 4. Общая удовлетворённость учебным материалом

4.1. Изучение материала вызвало устойчивый интерес к теме графического программирования.

Оценка: 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

4.2. Я готов(а) рекомендовать данный учебный материал другим студентам.

Оценка: 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

4.3. В целом я удовлетворён(а) качеством учебного пособия.

Оценка: 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

Дополнительные вопросы (открытая форма)

5.1. Какие модули курса показались вам наиболее полезными? Почему?

5.2. Какие темы вызвали наибольшие затруднения? Что бы вы предложили улучшить?

5.3. Ваши дополнительные пожелания и комментарии:

Благодарим вас за участие в апробации учебного материала!

Дата заполнения: «____» _____ 20____ г.

Подпись (по желанию): _____