

УДК 371.3

https://doi.org/10.33619/2414-2948/100/68

ЦИФРОВИЗАЦИЯ ОБУЧЕНИЯ ФИЗИКЕ: ИСПОЛЬЗОВАНИЕ КОМПЬЮТЕРНЫХ МОДЕЛЕЙ И АНИМАЦИИ

©**Аркабаев Н. К.**, ORCID: 0009-0000-1912-2225, SPIN-код: 9304-5193,
канд. физ.-мат. наук, Ошский государственный университет,

г. Ош, Кыргызстан, narkabaev@oshsu.kg

©**Абдугулова Г. С.**, ORCID: 0009-0005-8614-1908, SPIN-код: 9304-5193,

Ошский государственный университет,

г. Ош, Кыргызстан, gabdugulova@oshsu.kg

©**Кудуев А. Ж.**, ORCID: 0000-0002-3345-1364, SPIN-код: 8519-5251,

канд. техн. наук, Ошский государственный университет,

г. Ош, Кыргызстан, altynbek_kuduev@mail.ru

DIGITALIZATION OF PHYSICS TEACHING: USING COMPUTER MODELS AND ANIMATIONS

©**Arkabaev N.**, ORCID: 0009-0000-1912-2225, SPIN-code: 9304-5193, Ph.D.,
Osh State University, Osh, Kyrgyzstan, narkabaev@oshsu.kg

©**Abdugulova G.**, ORCID: 0009-0005-8614-1908, SPIN-code: 9304-5193,

Osh State University, Osh, Kyrgyzstan, gabdugulova@oshsu.kg

©**Kuduev A.**, ORCID: 0000-0002-3345-1364, SPIN-code: 8519-5251, Ph.D.,

Osh State University, Osh, Kyrgyzstan, altynbek_kuduev@mail.ru

Аннотация. Детально анализируются современные тенденции внедрения информационных технологий в процесс обучения физике — использование компьютерного моделирования и визуализации для повышения эффективности усвоения сложного физического материала учащимися. Актуальность работы обусловлена важностью применения инновационных цифровых средств обучения в условиях стремительной модернизации образования. На примере моделирования разряда конденсатора через резистор подробно разобрана методика создания математических компьютерных моделей с программной реализацией на языке Python и графической визуализацией с помощью Matplotlib. Рассмотрены различные подходы к компьютерной анимации (многоэкранный, фрагментарный, элементный, образный) для наглядного представления динамических систем. Приведены конкретные примеры анимации физических процессов. Подробно освещена поэтапная методика разработки интерактивных электронных плакатов для обучения физике. Сформулированы методические рекомендации по применению компьютерных моделей на разных типах уроков и этапах учебного процесса. Материалы статьи адресованы учителям физики, методистам и специалистам в области современных образовательных технологий. Дается методическое обоснование целесообразности внедрения информационно-коммуникационных технологий в обучение физике, анализируются различные подходы к компьютерному моделированию и визуализации, рассматриваются конкретные примеры и этапы создания анимационных и модельных пособий для школьников.

Abstract. The article analyzes current trends in the introduction of information technologies into the physics teaching process – the use of computer modeling and visualization to improve the efficiency of mastering complex physical material by students. The relevance of the work is due

to the importance of using innovative digital teaching tools in the context of the rapid modernization of education. Using the example of modeling the discharge of a capacitor through a resistor, the technique of creating mathematical computer models with software implementation in Python and graphical visualization using Matplotlib is examined in detail. Various approaches to computer animation (multi-screen, fragmentary, elemental, figurative) are considered for presenting dynamic systems. Specific examples of animation of physical processes are given. The systematic technique of developing interactive electronic posters for teaching physics is highlighted. Methodological recommendations for the use of computer models in different types of lessons and at different stages of the educational process are formulated. The materials of the article are addressed to physics teachers, methodologists and specialists in the field of modern educational technologies. The article provides a methodological rationale for the feasibility of introducing information and communication technologies into physics teaching, analyzes various approaches to computer modeling and visualization, and examines specific examples and stages of creating animation and modeling tools for schoolchildren.

Ключевые слова: моделирования, конденсатор, заряд, разряд, мультипликация, Python, Matplotlib.

Keywords: modeling, capacitor, charge, discharge, animation, Python, Matplotlib.

Современный этап развития общества характеризуется стремительным прогрессом в области информационных технологий. Цифровизация и компьютеризация проникают во все сферы человеческой деятельности, не обходят они стороной и систему образования. Использование новейших технических средств обучения открывает большие возможности для повышения эффективности учебного процесса, в том числе и в преподавании физики.

Одним из перспективных направлений применения информационных технологий в образовании является компьютерное моделирование физических явлений и процессов. Использование виртуальных лабораторий и интерактивных моделей позволяет визуализировать абстрактные процессы, сделать обучение более наглядным и доступным. Кроме того, анимация как способ представления динамических систем активно применяется при изучении физики в школах и вузах.

При моделировании сложных систем используются различные виды моделей. Наиболее мощным средством исследования, анализа и синтеза систем является математическая модель.

Математическое моделирование представляет собой уравнение или систему уравнений, связывающих независимые и зависимые величины [2].

С моделями можно проводить три вида эксперимента:

Одноразовый эксперимент, задается конкретное значение, вычисляется соответственно значение функции;

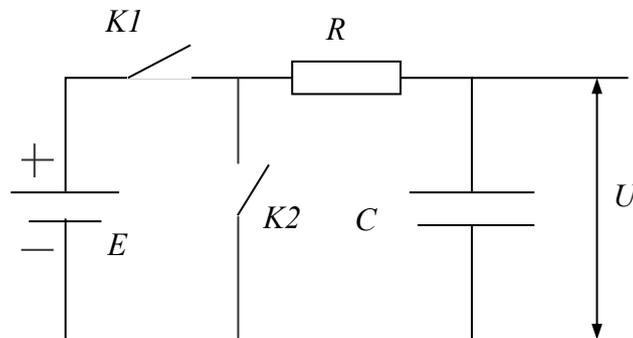
Многokrатное (автоматические) эксперименты в цикле, например от одного значения аргумента до другого с выбранным шагом;

Эксперимент со случайными значениями аргумента – имитационное моделирование (имитация - подражание).

Методика разработки математических моделей физических явлений и процессов

В качестве примера математического моделирования физических процессов возьмем процесс заряда и разряда конденсатора.

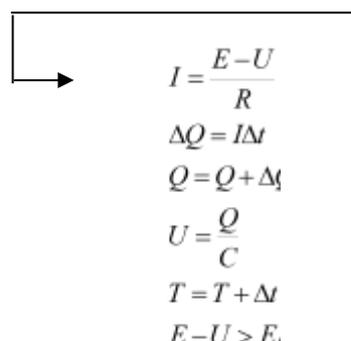
Из курса физики известно, что заряд конденсатора происходит по сложному экспоненциальному закону: $U = U_0(1 - e^{-\frac{t}{RC}})$, однако данная зависимость не всегда понятна учащимся. Поэтому метод Эйлера применительно к данному процессу позволяет сделать эту зависимость более понятной для учащихся, так как математическая модель представляется цепочкой простых и известных законов [1].



Нами предложено отображать математическую компьютерную модель изображать в виде неопределенного цикла, с многократным повторением вычислений по одной и той же цепочке уравнений. В чем-то это напоминает алгоритм Эйлера для решения дифференциальных уравнений.

Рассмотрим последовательность разработки компьютерной модели процесса заряда конденсатора заданной емкости C от источника э.д.с. E через резистор с известным сопротивлением R . Слева приведена алгоритмическая запись математической модели процесса заряда конденсатора.

Для того, чтобы модель заработала, вначале вводят исходные данные: $E, C, R, \varepsilon, \Delta t$, а также начальные значения величин: $t = 0, q = 0, U = 0$.



Ток через резистор R зависит от разности напряжений $E - U$ и поэтому вначале через него течет большой ток, а по мере заряда напряжение на обкладках конденсатора повышается и ток заряда уменьшается и процесс заряда замедляется. Цикл может повторяться бесконечно, так как теоретически конденсатор зарядить точно до напряжения $U = E$ невозможно, поэтому процесс заряда считается законченным, когда эта разность станет меньше наперед заданной бесконечно малой величины ε .

Аналогично выглядит математическая модель разряда конденсатора через резистор с заданным сопротивлением R . При этом ток через резистор вначале большой, он определяется напряжением на обкладках конденсатора, а в конце разряда напряжение на

конденсаторе приближается к нулю и процесс разряда замедляется и может продолжаться бесконечно. Так же, как и при заряде, процесс ограничивается моментом, когда напряжение на конденсаторе станет меньше наперед заданной величины ε .

Перед тем как начать работу с моделью, вводят исходные данные и начальные значения: $U, C, R, \varepsilon, \Delta t, t=0, q=CU$. Далее, в первую очередь, вычисляется ток. Затем, считая, что в течение короткого промежутка времени Δt значение тока не изменилось, подсчитывается количество перенесенного, с одной обкладки конденсатора на другую, заряда. Это – потерянный конденсатором заряд, поэтому он вычитается из текущего заряда. Зная заряд и емкость конденсатора, можно вычислить напряжение на конденсаторе, которое уменьшилось. Параллельно идет подсчет времени разряда, который прекращается, как только напряжение на конденсаторе достигнет значения $U < \varepsilon$. В противном случае цикл вновь повторяется, то есть подсчитывается ток, затем потерянный заряд, новое значение заряда и напряжения и так далее.

$$\begin{aligned} I &= \frac{U}{R} \\ \Delta Q &= I \Delta t \\ Q &= Q - \\ U &= \frac{Q}{C} \\ T &= T + \\ U &> \varepsilon \end{aligned}$$

Ниже приводится стержневой фрагмент математической модели разряда конденсатора на языке программирования Python:

```
import numpy as np
import matplotlib.pyplot as plt
class Capacitor:
    def __init__(self, C, R):
        self.C = C
        self.R = R

    def discharge(self, U0, t_max, dt):
        results = []
        t = 0
        U = U0
        while t <= t_max and U >= 1e-3*U0:
            I = U/self.R
            dQ = I*dt
            U_new = U - dQ/self.C

            results.append([t, U])

            U = U_new
            t += dt
```

```
return np.array(results)

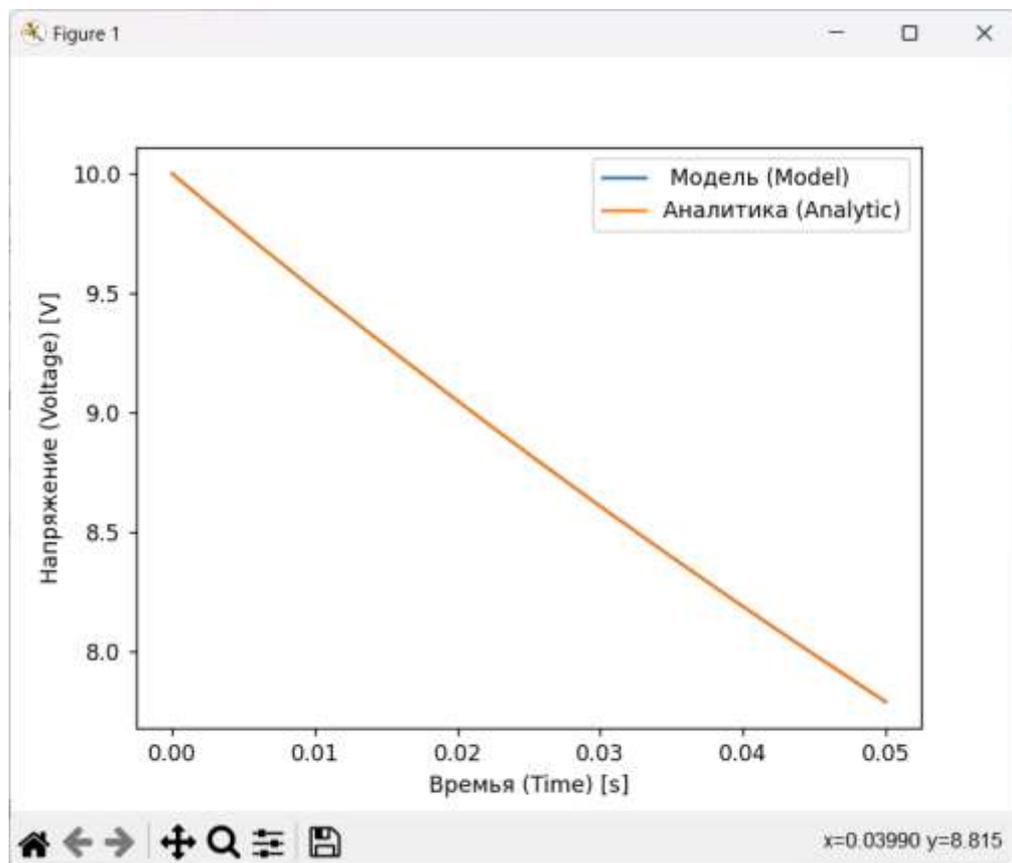
C = 10e-6
R = 20e3
U0 = 10
model = Capacitor(C, R)
results = model.discharge(U0, 0.05, 1e-5)

time = results[:,0]
voltage = results[:,1]

analytic_voltage = U0*np.exp(-time/(R*C))

plt.plot(time, voltage, label=' Модель (Model)')
plt.plot(time, analytic_voltage, label='Аналитика (Analytic)')
plt.legend()
plt.xlabel('Время (Time) [s]')
plt.ylabel('Напряжение (Voltage) [V]')
plt.show()
print('Время разряда (Discharge time):', time[-1], 's')
```

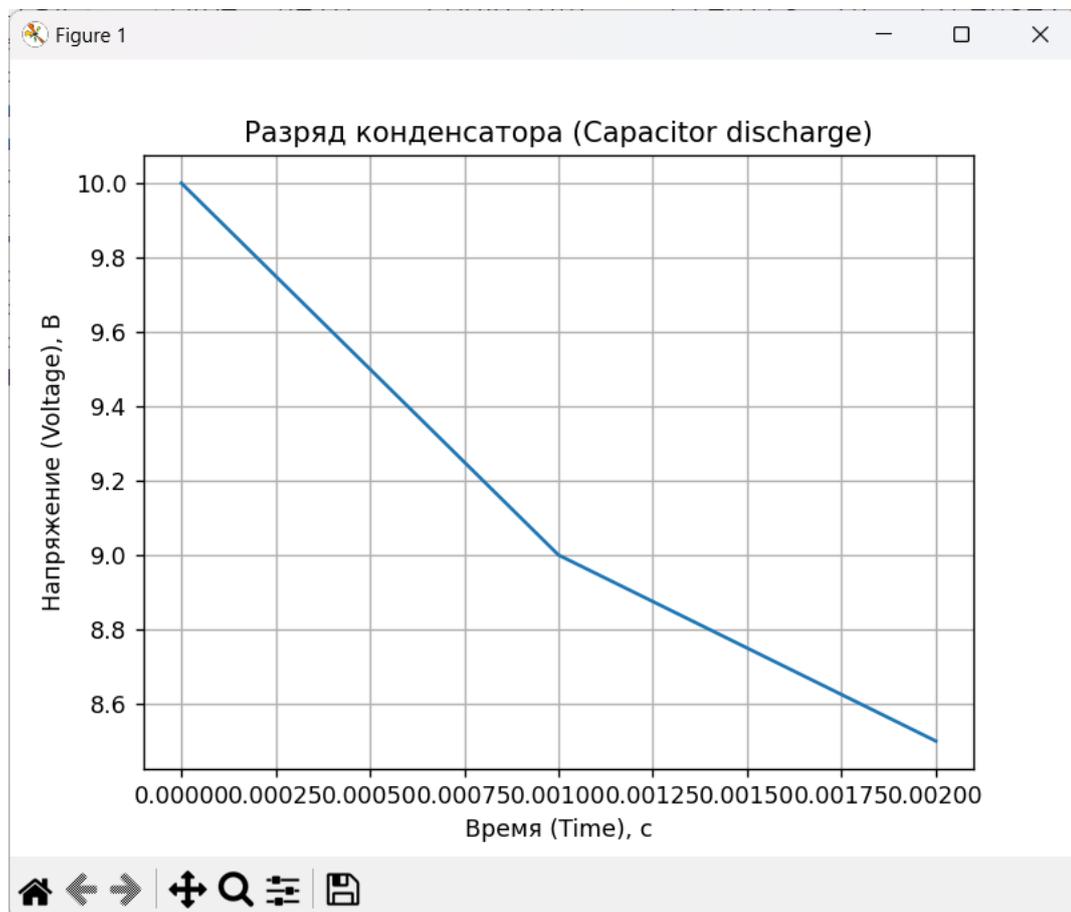
Результат данного кода получить следующий вид:



Данный подход можно распространить на моделирование других электрических цепей, используя объектно-ориентированный подход.

Для вывода результатов моделирования разряда конденсатора в виде графиков можно использовать библиотеку `matplotlib` и использовать различные стили и настройки для кастомизации графиков. Таким образом `matplotlib` предоставляет гибкие средства визуализации данных. В качестве примера рассмотрим следующий листинг программы:

```
import matplotlib.pyplot as plt
import numpy as np
time_array = [0.0, 0.001, 0.002]
voltage_array = [10.0, 9.0, 8.5]
plt.plot(time_array, voltage_array)
plt.xlabel('Время (Time), c')
plt.ylabel('Напряжение (Voltage), В')
plt.title("Разряд конденсатора (Capacitor discharge)")
plt.grid()
plt.savefig('discharge.png')
plt.show()
plt.plot(t1, u1, label='Модель (Model)')
plt.plot(t2, u2, label='Аналитика (Analytic)')
plt.legend()
```



Методы анимации изображений

Существует несколько методов получения подвижных изображений – анимации изображений [4]:

1) *Многоэкранный режим*. Некоторые компьютеры, например, Агат, имеют несколько одинаковых графических экранов. Если на этих экранах создать последовательность кадров, из которых каждый последующий кадр незначительно отличается от предыдущего, и воспроизвести их один за другим, то получится эффект движения, как в мультипликационном фильме. Если компьютер имеет 12-16 таких экранов, то, записав в них последовательность кадров, можно продемонстрировать короткий мультипликационный фильм. Обычно, последовательность кадров непрерывно повторяют, чтобы получить периодические движения, например, сжатие и расширение газа при движении поршня, периодические колебания маятника и др.

2) *Фрагментарный режим*. В большинстве современных компьютеров количество экранных режимов ограничено, причем экраны имеют разную разрешающую способность, то есть содержат разное число пикселей, и поэтому изображения на разных экранах не совместимы друг с другом. Для осуществления эффекта мультипликации каждый кадр из намеченной последовательности кадров рисуется на одном и том же экране, причем каждый кадр приходится рисовать заново. А это требует значительного времени и делает невозможным получение эффекта мультипликации.

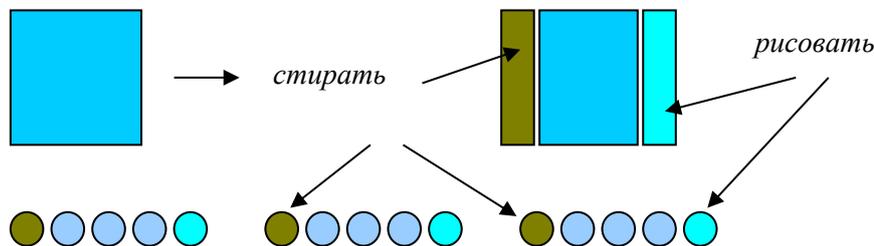
3) *Элементный режим* – движущийся объект представляется набором элементов (точек, линий), каждый из которых перемещается от кадра к кадру. Применяется при моделировании движения зарядов, распространения волн.

4) *Образный режим* – подвижный объект сохраняется в виде целостного графического образа, который затем переносится по экрану. Используется в компьютерных моделях движения тел, работы механизмов.

Образный режим в компьютерной анимации реализуется как движущийся объект представляемый целостный графический образ. Для него выделяется память, в которой хранится информация о графических элементах объекта. А образ представляет собой прямоугольную область, наиболее плотно охватывающую графическое изображение. Для идентификации используются координаты левого верхнего и правого нижнего углов этой области. Перемещение образа на экране осуществляется путём последовательного стирания и воспроизведения, графическая информация при этом считывается из заранее выделенной области памяти. Положение образа задается координатами левого верхнего угла. Перенос осуществляется сдвигом адресов памяти, что позволяет быстро организовать анимацию. Для восстановления фона используется логическая операция исключающего ИЛИ при наложении образа на изображение на предыдущих кадрах. Образный режим позволяет упростить процедуру анимации и обеспечить её высокую скорость при моделировании движения объектов.

Если для записи (изменения) информации об одном пикселе требуется 1 мкс времени, то на запись одного полного кадра размером 640×480 пикселей потребуется $640 \cdot 480 = 307200$ мкс, то есть 0,3 с, что составит частоту всего 3 кадра в секунду, в то время, как для мультипликации требуется частота не менее 16 кадров в секунду. Но если изменять содержимое не всего кадра, а только его отдельной части, например участка размером 100×100, на это потребуется $100 \cdot 100 = 10000$ мкс, то есть всего 0,01 с.

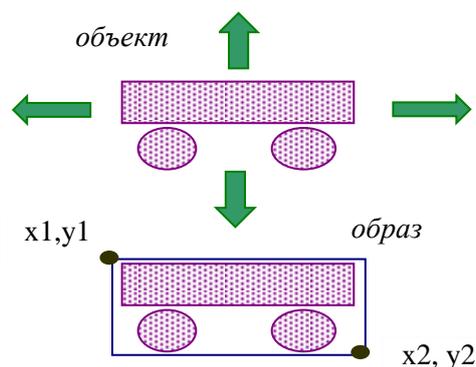
Таким образом, чтобы сократить время рисования кадра, заново рисуют только часть (фрагмент) изображения, оставляя остальное без изменения. Например, чтобы показать работу тепловой машины, заново рисуется подвижный поршень, а неподвижный цилиндр сохраняется неизменным в каждом кадре.



3) *Элементный режим.* Для еще более существенного сокращения времени создания очередного кадра при рисовании движущегося объекта, его рисуют не целиком, а только его отдельные элементы. Например, поступательно движение квадрата можно показать, рисуя узкую полоску впереди и одновременно стирая такую же полоску сзади. При этом квадрат, кадр за кадром, перемещается в одном направлении. Ширину полоски обычно берут равной одному пикселю. Движение зарядов вдоль проводника можно изобразить движущейся пунктирной линией, причем движение каждой черточка достигается рисованием одной точки впереди и стиранием одной точки сзади. Кадр за кадром черточки двигаются в одном направлении.

4) *Образный режим.* Если при перемещении по экрану форма объекта остается неизменной, то объект запоминается и сохраняется в памяти как один объект памяти, то есть запоминается образ объекта. Этот объект перемещается по экрану как значок и это достигается перемещением фрагментов памяти целиком на соответствующее число адресов.

Образ «запоминается» в виде прямоугольника, внутри которого целиком размещается перемещаемый объект. Эта область ограничена двумя вертикалями и двумя горизонталями, проходящими через начальную и конечную точки (x_1, y_1) и (x_2, y_2) . За начало образа берется верхняя левая точка, а конечной является нижняя правая точка этого прямоугольника.



Принцип перемещения образа на экране заключается в стирании образа на старом месте и воспроизведении его на новом месте, при этом перемещение происходит практически мгновенно. Размещение образа в нужном месте экрана производится по заданным координатам начальной точки.

Стереть, означает нарисовать объект цветом фона. Но иногда образ движется над изображением другого объекта и после прохождения нужно восстановить изображение

объекта, над которым прошел образ. Это достигается использованием логической операции ИСКЛЮЧАЮЩЕЕ ИЛИ, позволяющее восстановить рисунок фона после прохождения образа [1, 2].

Рассмотрим в конкретном примере визуализации физического процесса компьютерную модель движения маятника. Для этого нужно импортировать необходимые библиотеки как `numpy`, `matplotlib` и `animation`.

```
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.animation as animation
g = 9.8
L = 1
theta_0 = 60

fig = plt.figure()
ax = plt.axes(xlim=(-2, 2), ylim=(-2, 1))
line, = ax.plot([], [], lw=2)
ball, = ax.plot([], [], 'o', ms=12)
T = 2*np.pi*np.sqrt(L/g)

def animate(i):
    theta = theta_0*np.cos((i/50)*2*np.pi/T)
    x = L*np.sin(theta)
    y = -L*np.cos(theta)

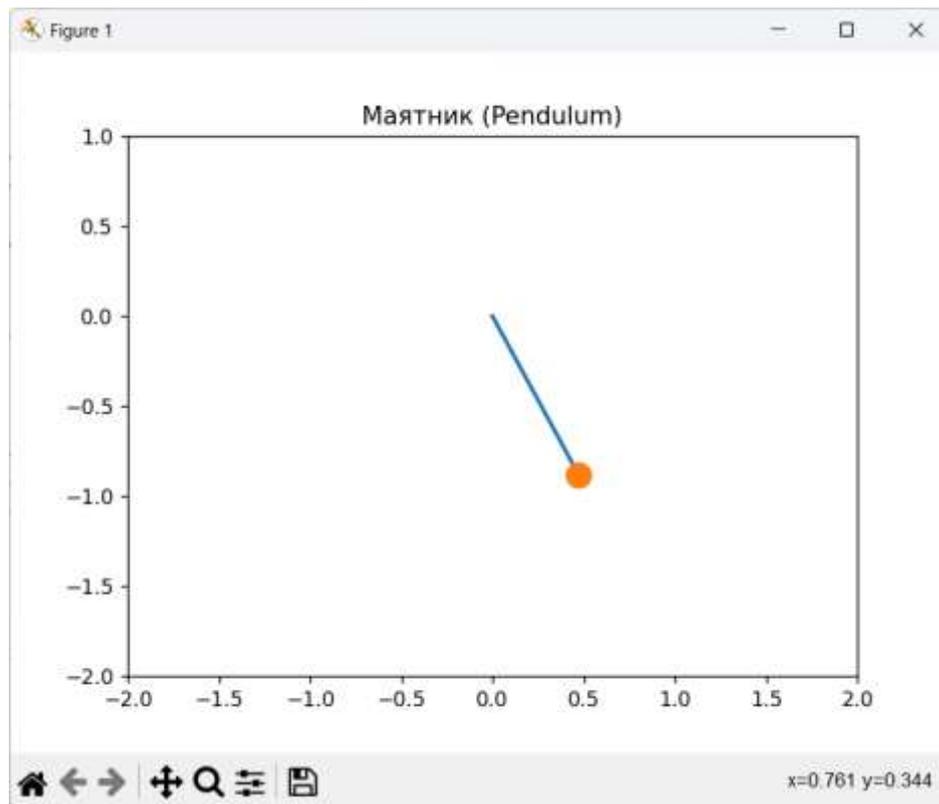
    line.set_data([0, x], [0, y])
    ball.set_data([x], [y])
    return line, ball

anim = animation.FuncAnimation(fig, animate,
                               frames=200, interval=20)
plt.title('Маятник')
plt.show()
```

Здесь рассчитан период на основе длины нити, учтен масштаб по времени для плавной анимации. На каждой итерации цикла маятник движется по координатам x , y центра, соответствующие движению по горизонтали и вертикали.

Библиотека `matplotlib` автоматически отображает кадры анимации с заданным интервалом.

Данный подход можно использовать для визуализации других различных физических процессов, добиваясь ощущения динамики и движения.



Методика создания электронных динамических контактов

Создание динамического плаката – целое искусство, эта работа интересна и увлекательна. Последовательность создания динамического компьютерного плаката представляет целый комплекс работ и требует от его создателя не только глубокого и тонкого знания соответствующего учебного материала, но и умений художника, чертежника, дизайнера и др. Мы выделили 10 основных этапов в этой работе [5]:

1. Постановка задачи. Создание эскиза объекта демонстрации с учетом размеров и пропорций выбранного графического экрана.

2. Определение основных стадий (этапов) развития процесса и составление эскизов каждого кадра. Если число кадров невелико (до 5-10), выбирается дискретный режим демонстрации, иначе выбирается режим мультипликации.

3. Определение направления и границ перемещения графических объектов на экране. При этом нужно проследить, чтобы изображение не выходило за рамки экрана. Выбор масштаба пересчета размеров объекта в экранные пиксели.

4. Определение координат опорных точек объектов (в пикселях) с учетом перемещения отдельных частей в процессе демонстрации всех кадров. Нанесение значений координат на эскиз.

5. Написание графической программы на выбранном алгоритмическом языке (Python). При этом рекомендуется использовать подпрограммы.

6. Отлаживание программы, включение в программу заглушек, ловушек с целью обнаружения и локализации ошибок. Постепенное включение и отладка подпрограмм и процедур, пока после снятия заглушек и ловушек программа не заработает безотказно.

7. Размещение в кадрах надписей и символов с помощью текстового экрана с использованием позиционирования печати символов.

8. Установка механизма смены кадров (вручную – с помощью клавиш, или автоматически – через заданный интервал времени. Установка скорости перемещения объектов на экране.

9. Дополнение текста программы комментариями, чтобы сделать программу удобной для чтения, понимания и совершенствования пользователем, например, другим учителем.

10. Составление документации по применению программы - динамического плаката - на уроке с указанием особенностей и способов управления плакатом. Это необходимо, чтобы любой учитель физики мог заранее знать возможности и особенности данного дидактического пособия и встроить его в свой урок.

Методические рекомендации по применению компьютерных моделей в зависимости от типа урока, этапа изучения материала и поставленных задач.

Методические рекомендации по использованию компьютерных моделей на уроках физики можно представить следующим образом:

На этапе изучения нового материала:

демонстрация моделей физических процессов (например, модели движения тел, модели колебаний) для изучения и визуализации явлений;

проведение виртуальных интерактивных экспериментов (например, в виртуальных лабораториях по электричеству, оптике);

использование тестирующих и контролирующих программ для первичного закрепления знаний.

На этапе закрепления материала:

выполнение учащимися виртуальных лабораторных работ;

самостоятельные исследования с помощью интерактивных моделей (изменение параметров, анализ зависимостей);

использование компьютерного тестирования для контроля усвоения знаний.

При повторении и обобщении:

систематизация знаний с помощью обучающих или контролирующих программ;

итоговое компьютерное тестирование по разделам физики;

использование электронных справочников и энциклопедий для повторения теории.

Такой дифференцированный подход позволит эффективно интегрировать компьютерные средства обучения в учебный процесс на разных этапах.

Применение информационных технологий открывает новые возможности для повышения эффективности обучения физике. Использование компьютерных моделей и виртуальных лабораторий целесообразно на всех этапах учебного процесса.

На уроках изучения нового материала моделирование помогает визуализировать абстрактные физические концепции, демонстрировать опыты, трудно реализуемые в реальных условиях. Интерактивные симуляции обеспечивают наглядность, способствуя лучшему пониманию и запоминанию материала.

Для закрепления знаний эффективно использование виртуальных лабораторных практикумов. Это позволяет отрабатывать практические умения и исследовательские навыки при работе с цифровыми измерительными инструментами и установками. Ученики могут проводить серии виртуальных экспериментов, изучая физические закономерности.

На обобщающем этапе дидактически целесообразно применение тестирующих программ и электронных справочников для систематизации и контроля знаний.

Автоматизированная проверка закрепляет изученный материал, выявляя пробелы в освоении курса физики.

Таким образом, в зависимости от этапа обучения подбираются определённые виды компьютерных средств, решающие конкретные методические задачи. Грамотное сочетание традиционных и инновационных образовательных технологий – залог высокой результативности процесса обучения физике.

Заключение

Таким образом, применение компьютерного моделирования на уроках физики имеет большой дидактический потенциал. Математические модели позволяют наглядно и доступно продемонстрировать различные физические процессы и явления. Компьютер позволяет визуализировать скрытые от глаз процессы, проводить виртуальные эксперименты.

Использование анимации делает процесс обучения более интересным и запоминающимся для учащихся. Различные методы компьютерной анимации – многоэкранный, фрагментарный, элементный и образный – расширяют возможности визуализации учебного материала.

Таким образом, внедрение информационных технологий в процесс обучения физике соответствует современным тенденциям развития образования. Компьютерные модели позволяют индивидуализировать процесс обучения, повышают мотивацию учащихся и способствуют лучшему усвоению сложного физического материала.

При разработке компьютерных моделей необходимо соблюдение дизайна, создание удобства управления: путем включения, остановки, пауз, повторного воспроизведения кадров, изменения условий эксперимента. Сказанное можно проиллюстрировать на примере создания динамического плаката «Работа судоводного шлюза», как пример к теме «Сообщающиеся сосуды», рассматриваемой по физике в 7 классе средней школы [3].

Список литературы:

1. Громов С. В. Физика: Оптика. Тепловые явления. Строение вещества. М.: Просвещение, 2002. 287с.
2. Гулд Х., Тобочник Я. Компьютерное моделирование в физике. Т. I-II. М.: Мир, 1990.
3. Каменецкий С. Е., Степанов С. В., Петрова Е. Б. Лабораторный практикум по теории и методике обучения физике в школе М.: Академия, 2002. 304 с.
4. Мastroпас З. П., Синдеев Ю. Г. Физика: Методика и практика преподавания. Ростов н/Д: Феникс, 2002. 288 с.
5. Аркабаев Н. К., Кудуев А. Ж., Сулайманов А. А. Обучение языка Python в школе: проблемы и эффективные методы // Вестник ошского государственного университета. Педагогика. Психология. 2023. №1(2). С. 24-29. [https://doi.org/10.52754/16948742_2023_1\(2\)_3](https://doi.org/10.52754/16948742_2023_1(2)_3)

References:

1. Gromov, S. V. (2002). Fizika: Optika. Teplovye yavleniya. Stroenie veshchestva. Moscow. (in Russian).
2. Guld, Kh., & Tobochnik, Ya. (1990). Komp'yuternoe modelirovanie v fizike. Moscow. (in Russian).
3. Kamenetskii, S. E., Stepanov, S. V., & Petrova, E. B. (2002). Laboratornyi praktikum po teorii i metodike obucheniya fizike v shkole Moscow. (in Russian).
4. Mastropas, Z. P., & Sindeev, Yu. G. (2002). Fizika: Metodika i praktika prepodavaniya. Rostov-on-Don. (in Russian).

5. Arkabaev, N. K., Kuduev, A. Zh., & Sulaimanov, A. A. (2023). Obuchenie yazyka Python v shkole: problemy i effektivnye metody. *Vestnik oshskogo gosudarstvennogo universiteta. Pedagogika. Psikhologiya*, (1(2)), 24-29. (in Russian). [https://doi.org/10.52754/16948742_2023_1\(2\)_3](https://doi.org/10.52754/16948742_2023_1(2)_3)

Работа поступила
в редакцию 15.02.2024 г.

Принята к публикации
21.02.2024 г.

Ссылка для цитирования:

Аркабаев Н. К., Абдугулова Г. С., Кудуев А. Ж. Цифровизация обучения физике: использование компьютерных моделей и анимации // Бюллетень науки и практики. 2024. Т. 10. №3. С. 520-532. <https://doi.org/10.33619/2414-2948/100/68>

Cite as (APA):

Arkabaev, N., Abdugulova, G., & Kuduev, A. (2024). Digitalization of Physics Teaching: Using Computer Models and Animations. *Bulletin of Science and Practice*, 10(3), 520-532. (in Russian). <https://doi.org/10.33619/2414-2948/100/68>