



Инженерный класс
В МОСКОВСКОЙ ШКОЛЕ

Компетентный подход на уроках информатики в предпрофессиональных классах технологического профиля

Попов Владислав Сергеевич, эксперт ИРПО МГПУ

1 Формирование прикладных профессиональных навыков (человек умелый).
Обеспечение усвоения знаний, информации, продуктов культуры (человек эрудированный).
Развитие общих способностей, ценностей, социальных умений (человек развитый).

2 Компетентность = знания + навыки + деятельностные установки → действие
И. А. Зимняя: + регуляция + готовность

3 Рационально использовать термин **«универсальные компетентности»** вместо «навыки XXI века», «гибкие навыки», «софт скиллс», «универсальные учебные действия»

Универсальные компетентности и новая грамотность: от лозунгов к реальности / под ред. М. С. Добряковой, И. Д. Фрумина ; при участии К. А. Баранникова, Н. Зиила, Дж. Мосс, И. М. Реморенко, Я. Хаутамяки ; Нац. исслед. ун-т «Высшая школа экономики». — М.: Изд. дом Высшей школы экономики, 2020. — 472 с. — 500 экз. — ISBN 978-5-7598-2177-9 (в обл.). — ISBN 978-5-7598-2074-1 (e-book).

1

Универсальные компетентности могут быть представлены как три блока навыков (три универсальные компетентности).

— Компетентность познания (мышления) (использование навыков мышления для решения интеллектуальных задач).

— Компетентность взаимодействия с другими людьми.

— Компетентность взаимодействия с собой (управление собой)

2

Ключевым вопросом развития универсальных компетентностей является **вопрос соотношения с предметными (дисциплинарными) знаниями и навыками**

3

Знания и навыки, включаемые в «новую грамотность», относятся либо к общей грамотности, либо к предметной

Универсальные компетентности и новая грамотность: от лозунгов к реальности / под ред. М. С. Добряковой, И. Д. Фрумина ; при участии К. А. Баранникова, Н. Зиила, Дж. Мосс, И. М. Реморенко, Я. Хаутамяки ; Нац. исслед. ун-т «Высшая школа экономики». — М.: Изд. дом Высшей школы экономики, 2020. — 472 с. — 500 экз. — ISBN 978-5-7598-2177-9 (в обл.). — ISBN 978-5-7598-2074-1 (e-book).

Доклад «Универсальные компетентности» vs И. А. Зимняя vs DeSeCo

«Универсальные компетентности»

- Компетентность познания (мышления) (использование навыков мышления для решения интеллектуальных задач).
- Компетентность взаимодействия с другими людьми.
- Компетентность взаимодействия с собой (управление собой).

И. А. Зимняя

- Компетенции и компетентности:
- Относящиеся к самому человеку как к личности, как субъекту жизнедеятельности, деятельности, общения.
 - Относящиеся к социальному взаимодействию человека с другими людьми.
 - Относящиеся к деятельности человека.

DeSeCo

- Using tools interactively, e.g. language, technology.
- Interacting in heterogeneous groups.
- Act autonomously.

Включает 8 ключевых компетенций

- Языковая грамотность. Communication in the mother tongue (2006) → Literacy competence (2018).
- Мультиязыковая компетенция. Communication in foreign languages (2006) → Multilingual competence (2018).
- Математические, научные, технологические, инженерные компетенции. Mathematical competence and basic competences in science and technology (2006) → Mathematical competence and competence in science, technology and engineering (2018).
- Цифровая компетенция. Digital competence (2006) → Digital competence (2018).
- Персональная, социальная, образовательная компетенция. Learning to learn (2006) → Personal, social and learning to learn competence (2018).
- Гражданская компетенция. Social and civic competences (2006) → Citizenship competence (2018).
- Предпринимательская компетенция. Sense of initiative and entrepreneurship (2006) → Entrepreneurship competence (2018).
- Компетенция в области культурной осведомлённости и выражения. Cultural awareness and expression (2006) → Cultural awareness and expression competence (2018).



1 Результаты базового/углублённого уровня изучения информатики ориентированы на: общую функциональную грамотность, получение компетентностей для повседневной жизни и общего развития / получение компетентностей для последующей профессиональной деятельности как в рамках данной предметной области, так и в смежных с ней областях.

2 Основная цель: обеспечение дальнейшего развития информационных компетенций обучающегося, его готовности к жизни в условиях развивающегося информационного общества и возрастающей конкуренции на рынке труда.

3 Описан планируемый результат обучения информатике в школе.



Попов В. С., Алефиренко Е. А., Черницына Л. Ю. Компетентностный подход в федеральных рабочих программах по информатике на уровне среднего общего образования // Сборник трудов XXII Открытой всероссийской конференции Преподавание информационных технологий в Российской Федерации – 2024. Тверь.

Кейс № 1: цифровая схемотехника

Проблема: Недостаточное освоение основ цифровой схемотехники.

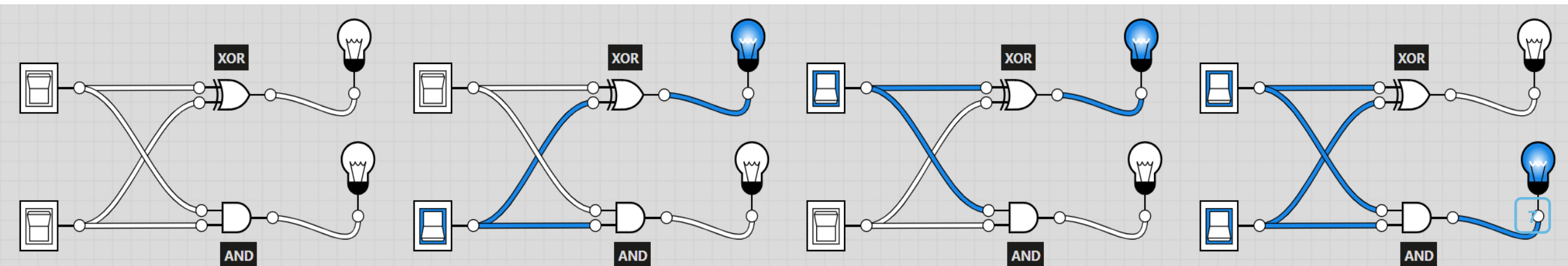
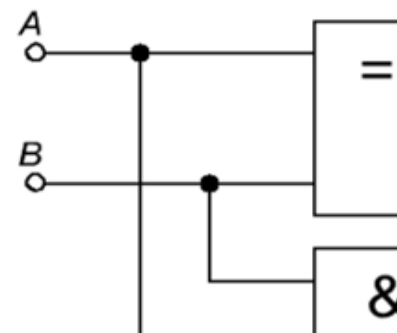
Решение: Лабораторная работа «Цифровая схемотехника и алгебра логики: создание цифровых устройств в онлайн-симуляторе».

Проводится в МГТУ с 2021 года

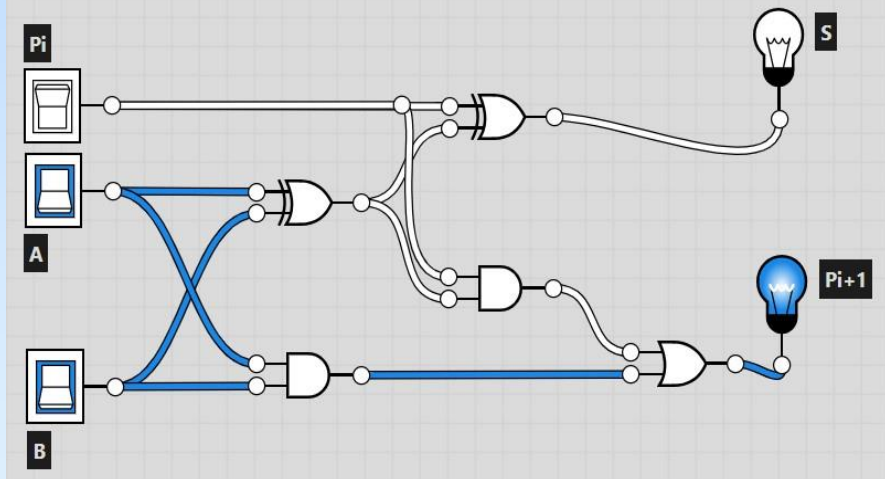
Время занятия: 90 минут

Классы: 10-11

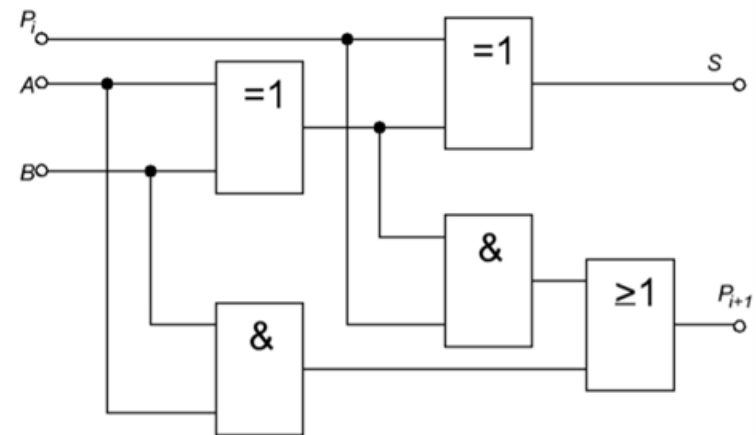
Полусумматор



Сумматор

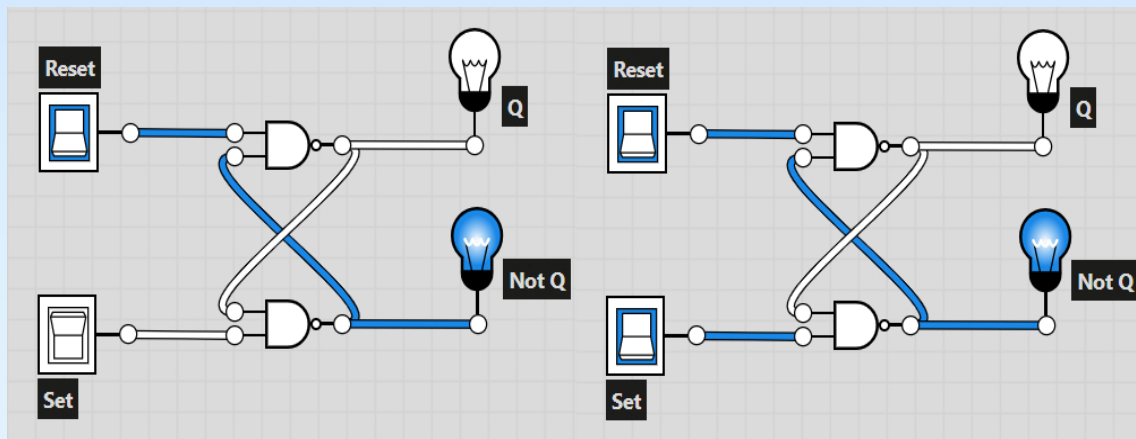


A	B	P_i	S	P_{i+1}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
1	0	0	1	0
0	1	1	0	1
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



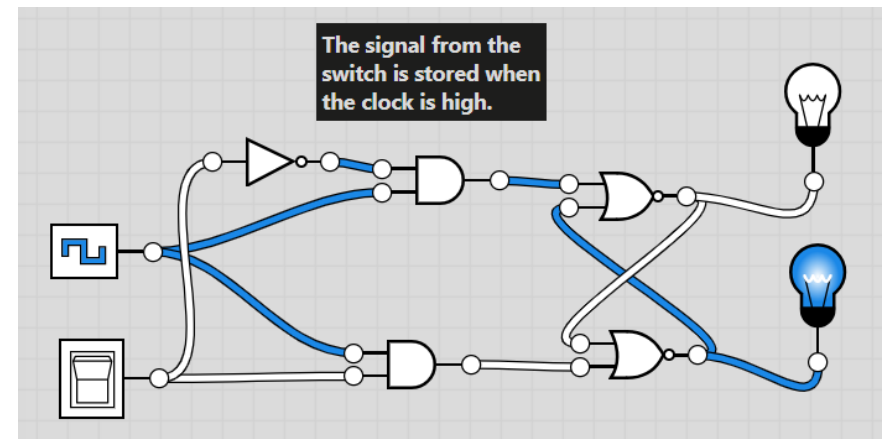
RS-триггер

Reset	Set	Q	$\neg Q$
0	0	Запрещено	
0	1	1	0
1	0	0	1
1	1	Хранение бита	



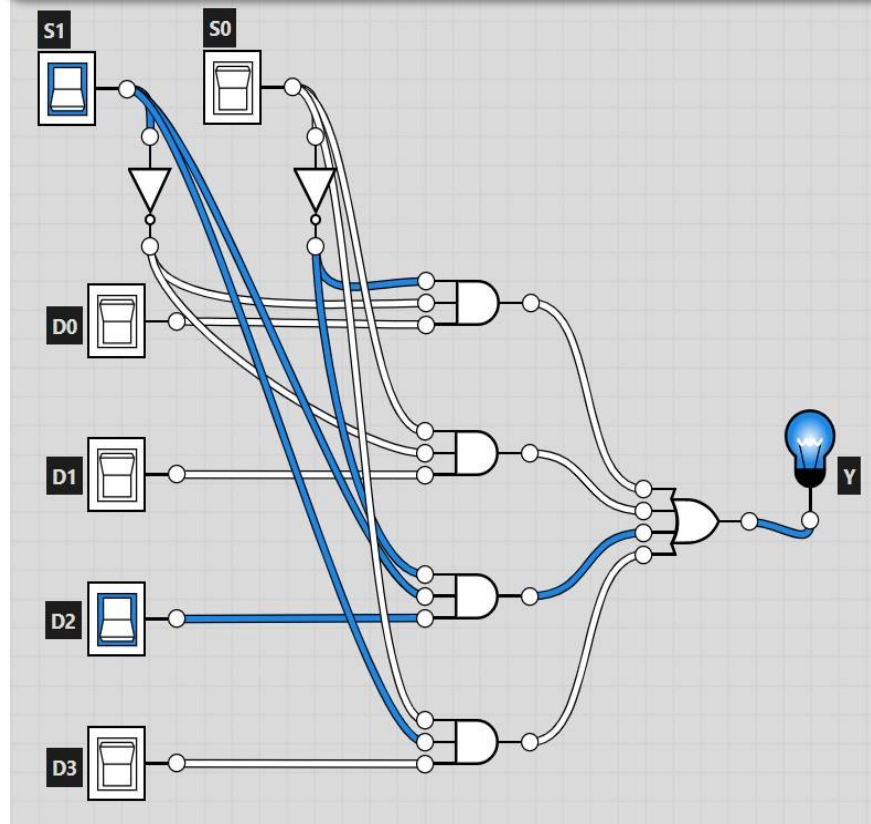
D-триггер

C	D	Q(t)	Q(t+1)	Комментарий
0	X	0	0	Режим хранения информации
0	X	1	1	
1	0	X	0	Режим записи информации
1	1	X	1	



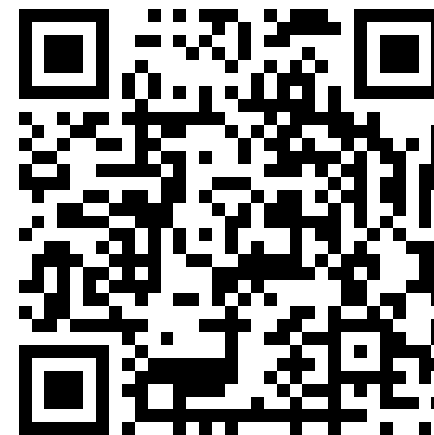
«Русский метод обучения ремёслам» – первая отечественная научная система начального профессионального трудового обучения, получившая мировое признание на всемирных выставках Вены, Парижа, Филадельфии второй половины XIX века. Следует отметить, что «русский метод обучения ремёслам», предложенный великим инженером и педагогом Д. К. Советкиным, по своей природе являлся компетентностно-ориентированным методом, совмещающим теорию и практику, основанным на упражнениях под контролем мастера и направленным на глубокое знание главных элементов профессиональной деятельности, способность и возможность эффективной практической деятельности.

Мультиплексор



Декомпозиция: знания

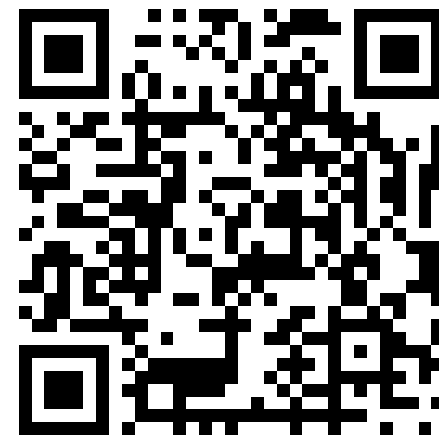
- **Предметные:** логические операции (включая редко рассматриваемые операции «исключающее или», «штрих Шеффера», «стрелка Пирса»), логические вентили (NOT, AND, OR, NAND, XOR), графическое представление логических вентилях, таблица истинности, полусумматор, сумматор, RS-триггер, D-триггер, мультиплексор, дешифратор, статическая память, полнота систем булевых функций, положительная логика.
- **Межпредметные:** последовательное и параллельное соединение элементов электрической цепи, интернациональные названия логических операций и вентилях, арность операций и функций, основы формальной логики.
- **Метапредметные:** информационные процессы (на примере цифровых устройств обработки, хранения, передачи информации), табличная и графическая формы представления информации, основы знаковых систем и семиотики; метапредметная знаниевая компонента может быть выражена через изучение перечисленных универсальных межпредметных понятий.



Попов В. С. Лабораторная работа «Цифровая схемотехника и алгебра логики: создание цифровых устройств в онлайн-симуляторе»: компетентностный подход в целях преемственности уровней образования // Информатика в школе № 2 (23), 2024. DOI: 10.32517/2221-1993-2024-23-2-35-43. С. 35-43.

Декомпозиция: навыки

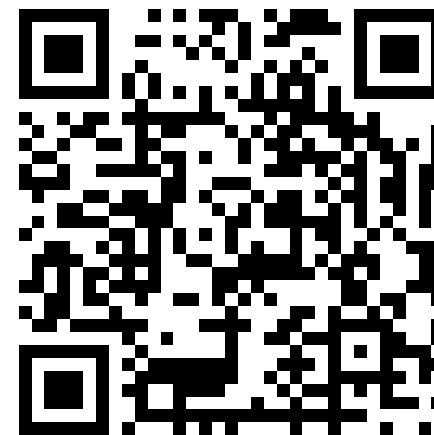
- *Предметные:* чтение таблицы истинности, чтение графической схемы цифрового устройства, применение логических операций и вентилей, синтез, моделирование, разработка и тестирование цифровых логических схем с использованием симулятора на основе таблиц истинности.
- *Метапредметные:* соотнесение табличной и графической форм представления информации (таблиц истинности и схем устройств цифровой схемотехники), межвидовое преобразование графической информации (изменение схемы стандарта ГОСТ/IEC на схему ANSI), комбинаторные навыки при создании и изучении таблицы истинности и цифровой логической схемы, применение онлайн-инструментов и модели SaaS в деятельности, развитие способностей моделирования, анализа, системного мышления, применения формальной логики, изобретательности и исследования моделей и идей.



Попов В. С. Лабораторная работа «Цифровая схемотехника и алгебра логики: создание цифровых устройств в онлайн-симуляторе»: компетентностный подход в целях преемственности уровней образования // Информатика в школе № 2 (23), 2024. DOI: 10.32517/2221-1993-2024-23-2-35-43. С. 35-43.

Декомпозиция: ценностные отношения

- Деятельностные установки / ценностные отношения: формирование готовности к пониманию и созданию схем устройств цифровой схемотехники, применению изученных основ алгебры логики для решения практических задач, развитие мотивации к получению дальнейшего образования в области цифровой схемотехники, цифровых автоматов, микропроцессорной техники, информационных технологий и инженерии, к эффективному применению средств ИКТ в практической деятельности.



Попов В. С. Лабораторная работа «Цифровая схемотехника и алгебра логики: создание цифровых устройств в онлайн-симуляторе»: компетентностный подход в целях преемственности уровней образования // Информатика в школе № 2 (23), 2024. DOI: 10.32517/2221-1993-2024-23-2-35-43. С. 35-43.

- ▶ Показан пример декомпозиции компетентностей на знания, навыки, деятельностные установки при изучении темы «Рекурсия».
- ▶ Показана необходимость и возможность формирования компетентности на различных уровнях сложности учебного материала, а также дифференциация формируемых компетентностей по уровням.
- ▶ Природа компетентности сама по себе рекурсивна.
- ▶ Для эффективного выполнения заданий на рекурсию и изучения темы необходимо формирование не отдельной компетентности, но суперпозиции компетентностей.

Математически описанная рекуррентная формула

$$\begin{aligned} F(n) &= 1 \text{ при } n = 1; \\ F(n) &= n + F(n - 1), \text{ если } n - \text{ чётно,} \\ F(n) &= 2 \times F(n - 2), \text{ если } n > 1 \text{ и при этом } n - \text{ нечётно.} \end{aligned}$$

Рекурсивная функция на языке программирования

```
def F(n):  
    if n == 1:  
        return 1  
    if n % 2 == 0:  
        return n + F(n - 1)  
    if n > 1 and n % 2 == 1:  
        return 2 * F(n - 2)
```



Задание: запишите четыре младших разряда тысячного числа Фибоначчи.

Вывод — 209-значное число:

434665576869374564356885276750406258025
646605173717804024817290895365554179490
518904038798400792551692959225930803226
347752096896232398733224711616429964409
065331879382989696499285160037044761377
95166849228875

На слайде: реализация мемоизации с использованием встроенных функций.

```
Python
import sys
sys.setrecursionlimit(5000)
from functools import lru_cache
@lru_cache(None)
def fib(n):
    if n == 0:
        return 0
    elif n == 1:
        return 1
    else:
        return fib(n - 1) + fib(n - 2)
print(fib(1000))
```

Задание: запишите четыре младших разряда тысячного числа Фибоначчи.

Вывод — 209-значное число:

434665576869374564356885276750406258025
646605173717804024817290895365554179490
518904038798400792551692959225930803226
347752096896232398733224711616429964409
065331879382989696499285160037044761377
95166849228875

На слайде: реализация мемоизации с использованием списка.

```
Python
N = 1000
mem = [0] * (N + 1)
def fib(n):
    if n == 0:
        return 0
    elif n == 1:
        return 1
    elif mem[n] != 0:
        return mem[n]
    else:
        mem[n] = fib(n - 1) + fib(n - 2)
        return mem[n]
print(fib(N))
```

Кейс № 3: углублённые задания на рекурсию

Компоненты Уровни	Знания (knowledge)	Навыки (skills)	Ценностные отношения (деятельностные установки, attitudes)
Базовый (Basic)	Стандартные модули <code>functools</code> и <code>sys</code> , декоратор <code>@lru_cache()</code> , функция <code>sys.setrecursionlimit()</code> , знание базовых примеров рекурсии и последовательностей, знание синтаксиса языка программирования для определения рекурсивной функции	Эпизодическое применение стандартных библиотечных возможностей кэширования и увеличения глубины по образцу, подстановка конкретных значений в рекуррентные формулы	Способность и готовность использования готовой библиотечной функциональности и базовых программ, использующих рекурсию для решения простых задач, способность повторного использования программного кода
Повышенный (Intermediate)	Понимание глубины рекурсии и возможности переполнения стека вызовов, прямой и косвенной рекурсии, прямого и обратного хода рекурсии, возможности замены рекурсивного алгоритма на циклический, знание и понимание терминов: рекурсия, рекурсивная функция, индуктивное определение, понимание принципа «разделяй и властвуй»	Построение графа вызовов рекурсивных функций, определение порядка вызовов рекурсивных функций, навык программирования рекурсивных алгоритмов, навык аналитического выполнения прямого и обратного хода рекурсии, способность взаимозамены рекурсивных и циклических алгоритмов	Способность использования рекурсивных алгоритмов для решения задач, их применения в широком контексте проблем и явлений, позитивное ценностное отношение к рекурсии как универсальному способу решения задач, лежащему в основе компьютерных вычислений
Углублённый (Advanced)	Виды кэширования и их особенности, виды рекурсивных функций и их особенности, знание тезиса Чёрча-Тьюринга, проблем останковки и разрешимости, временной сложности, длинной арифметики	Навык самостоятельной организации кэширования значений функций одной и нескольких переменных с использованием структур данных	Способность и готовность к самостоятельной разработке и исследованию сложных алгоритмов, практическому применению теории сложности вычислений

Цель: эмпирическое исследование времени выполнения рекурсивных функций и развитие STEM-компетенций.

1 Рекурсивная функция и тайминг кода

Пример вывода:

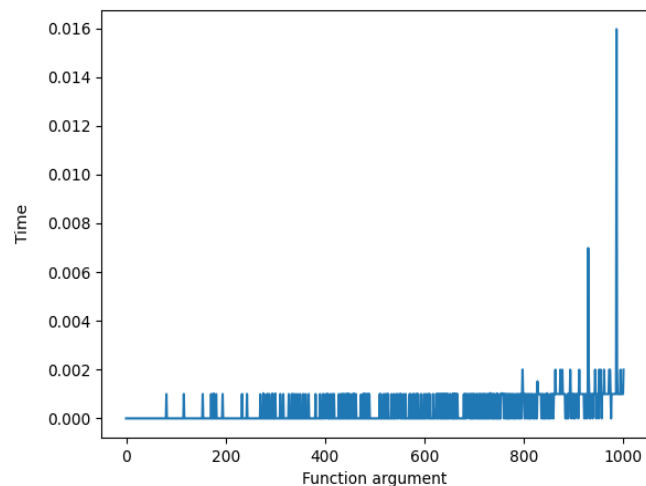
```
0.0000000000000000 27 10888869450418352160768000000
0.0000000000000000 28 304888344611713860501504000000
0.000997304916382 29 8841761993739701954543616000000
0.0000000000000000 30 26525285981219105863630848000000
0.0000000000000000 31 822283865417792281772556288000000
```

Python

```
import time
def fact(n):
    if n == 0:
        return 1
    if n > 0:
        return n * fact(n - 1)
for i in range(1001):
    start_time = time.time()
    f = fact(i)
    end_time = time.time()
    print('%.15f' % (end_time - start_time), i,
f)
```

Цель: эмпирическое исследование времени выполнения рекурсивных функций и развитие STEM-компетенций.

2 Рекурсивная функция и тайминг кода

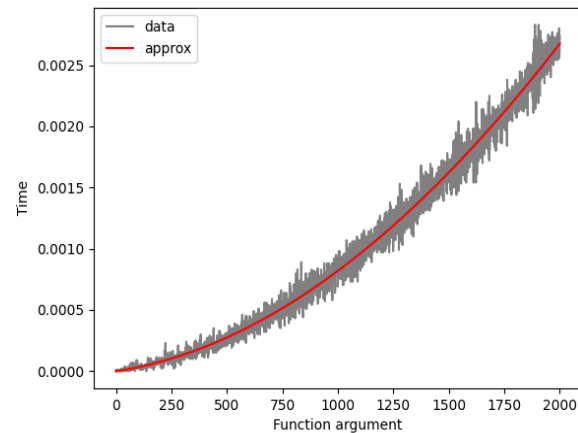
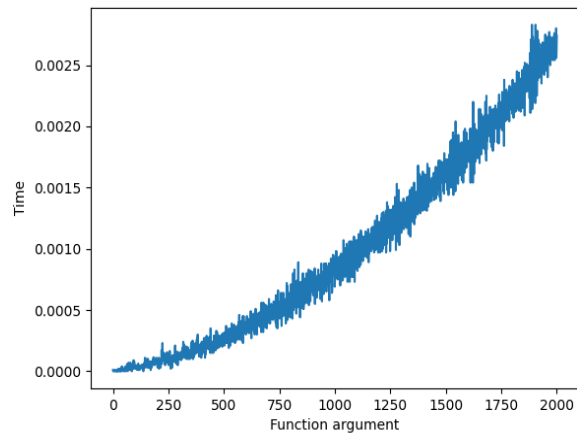


```
Python

import time
import matplotlib.pyplot as plt
def fact(n):
    if n == 0:
        return 1
    if n > 0:
        return n * fact(n - 1)
lst = []
for i in range(1001):
    start_time = time.time()
    f = fact(i)
    end_time = time.time()
    lst.append(end_time - start_time)
    print('%.15f' % lst[-1], i, f)
plt.plot(lst)
plt.xlabel("Function argument")
plt.ylabel("Time")
plt.show()
```

Цель: эмпирическое исследование времени выполнения рекурсивных функций и развитие STEM-компетенций.

3 Усреднение и аппроксимация



Python

```
...
N = 2001
lst = [0] * N
N_iterations = 1000
for k in range(N_iterations):
    for i in range(N):
        start_time = time.time()
        f = fact(i)
        end_time = time.time()
        lst[i] += (end_time - start_time) /
N_iterations
    if k % 10 == 0:
        print(k)
plt.plot(lst)
plt.xlabel("Function argument")
plt.ylabel("Time")
plt.show()
print(*lst)
```

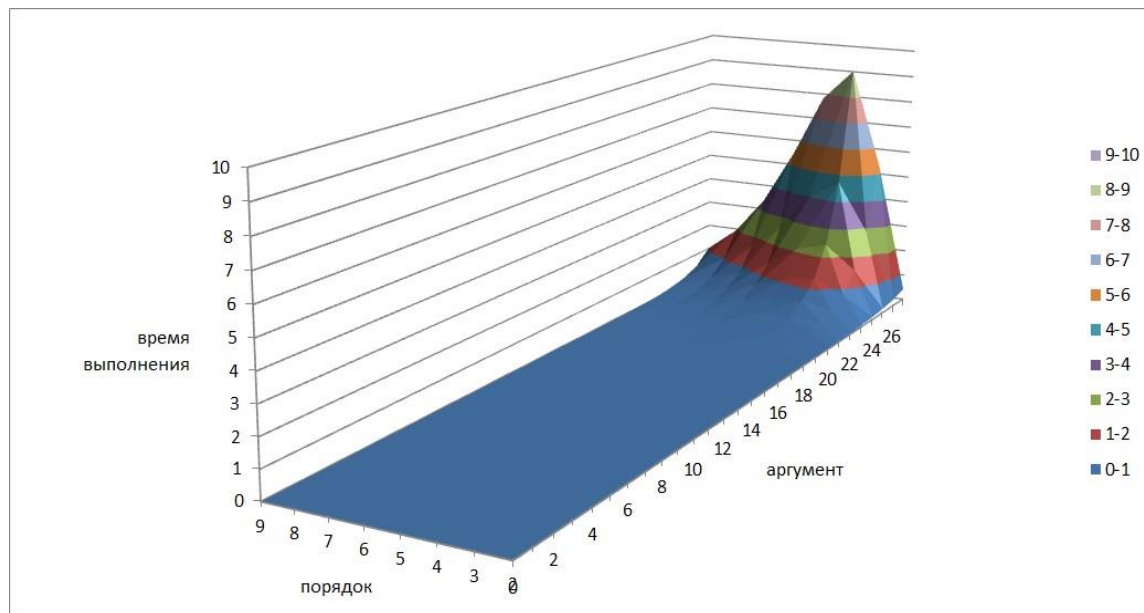
Цель: эмпирическое исследование времени выполнения рекурсивных функций и развитие STEM-компетенций.

4 DRY: обобщение функций Фибоначчи порядка n

```
Python
def fibn(n, order):
    if n < order - 1:
        return 0
    if n == order - 1:
        return 1
    else:
        sum = 0
        for i in range(order):
            sum += fibn(n - i - 1, order)
        return sum
```

Цель: эмпирическое исследование времени выполнения рекурсивных функций и развитие STEM-компетенций

5 Построение трёхмерного графика



Python

```
for k in range(N_iterations):  
    for order in range(2, 10):  
        for i in range(28):  
            start_time = time.time()  
            f = fibn(i, order)  
            end_time = time.time()  
            A[order][i] += (end_time -  
start_time) / N_iterations
```

Цель: эмпирическое исследование времени выполнения рекурсивных функций и развитие STEM-компетенций

Science Tech Eng Math

→ рекурсия, усреднение по ансамблю, линейный и полиномиальный рост, вычислительная сложность, аппроксимация, научный эксперимент, обработка и анализ его результатов, научная гипотеза

→ рекурсивные функции, циклы и вложенные циклы, тайминг кода (вычисление времени выполнения фрагмента программного кода) и его усреднение, визуализация данных (на примере построения графиков), длинная арифметика, реализация аппроксимации, одномерные и двумерные списки, принцип программирования DRY

→ функция вычисления факториала и обобщённые функции Фибоначчи различного порядка – от трибоначчи до ноначчи, – аппроксимирующая функция, усреднение, 2D- и 3D-графики функций

- ▶ Перечислены навыки, знаниевые, ценностные компоненты для базового и углублённого уровней изучения.
- ▶ Компоненты математической компетентности:
перевод чисел из одной системы счисления в другую, счёт в различных системах счисления, математические операции, определение чётности числа, значащие и незначащие разряды, применение математического индуктивного метода.
- ▶ Для эффективного выполнения заданий снова необходимо формирование не отдельной компетентности, но суперпозиции компетентностей.



Попов В. С. Взаимодействие информационных и математических компетенций и компетентностей на примере решения задания на IP-адресацию // Математика, информатика, физика: проблемы и перспективы. Международная научно-практическая конференция: сборник научных статей международной научно-практической конференции «Математика, информатика, физика: проблемы и перспективы», Оренбург, 25–26 апреля 2024 г. / ответственный редактор доктор педагогических наук И.В. Игнатушина; Министерство просвещения РФ; ФГБОУ ВО «Оренбургский государственный педагогический университет». – Оренбург: [б.и.], 2024. – 702 с.

Контакты

Проектная группа «Информатика в предпрофессиональном образовании»

 <https://profil.mos.ru>

 alefirenkoea@mgpu.ru





Инженерный класс
В МОСКОВСКОЙ ШКОЛЕ

Компетентностный подход на уроках информатики в предпрофессиональных классах технологического профиля

Попов Владислав Сергеевич, эксперт ИРПО МГПУ