



Урок 8. Python: Построение и проверка графика

Часть 1. Материалы ученика

Программирование на Python и C++ · 9 класс

Урок 8 из 34 · 45 минут · Модуль «Графики»

Уровень: начинающий программист; Python 3.

Программа: <https://schoolotzyv.ru/blog/python-cpp-9-klass-programma>

График должен показывать ровно полученные точки. Соединяющая линия помогает читать последовательность, но не доказывает, что между измерениями происходило именно это.

Результат урока: Построить PNG по проверенным парам, различить маркеры и соединяющую линию, проверить крайние и пустые серии.

01. Проблемная ситуация

0–5 минут. Учебная серия (0,2),(1,4),(4,8). Предскажите высоту среднего маркера и наклон обоих отрезков. Можно ли назвать высоту в минуту 2 измеренным значением?

02. Понятия и разобранный пример

5–14 минут: 4 минуты на понятия, 3 на пример, 2 на объяснение способа партнёру.

Маркеры представляют данные

`plot(x,y,"o-")` ставит маркеры и соединяет соседние пары в переданном порядке. При `x=[0,1,4]` интервалы на рисунке различны. Наклон первого отрезка $(4-2)/(1-0)=2$ см/мин, второго $(8-4)/(4-1)=4/3$ см/мин. Рост высоты за весь опыт 6 см не равен каждому локальному наклону.

Линия между точками — выбранный способ показа, а не дополнительное измерение. Для отдельных наблюдений без модели промежутков можно оставить только маркеры. Пустой список не соединяется и подписывается No data; единственная точка остаётся видимым маркером.

Контракт тот же: до 20 пар целых, $0 \leq t \leq 4$, $0 \leq h \leq 10$, строго возрастающие времена. Полный код проверяет диапазон и порядок до рисования. Фиксированные оси позволяют сравнивать несколько серий; автоматические границы могут визуально преувеличивать малые изменения.

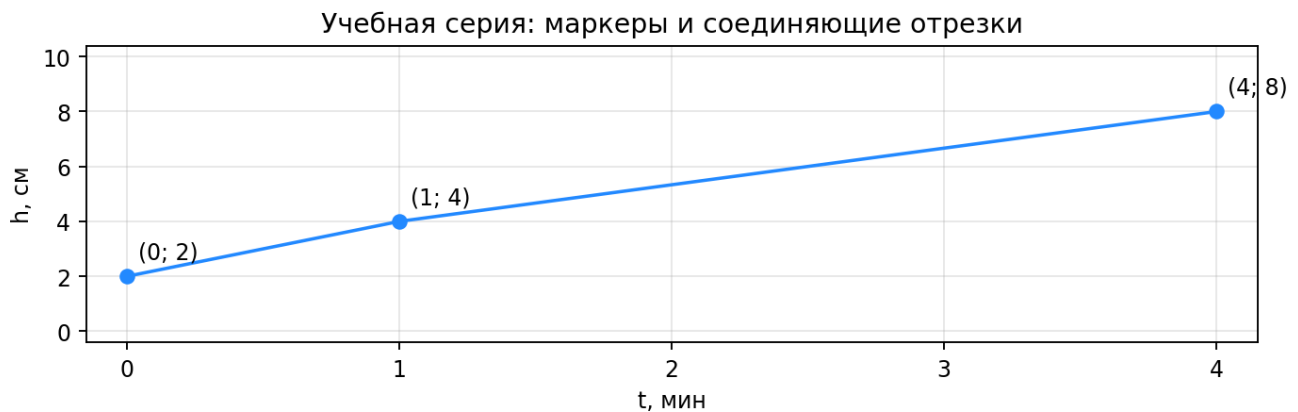
Отступы набирайте обычными пробелами: 4 на уровень. При копировании с веб-страницы замените неразрывные пробелы обычными; полный код с обычными пробелами есть в PDF.

Для основной серии проверяем три маркера (0,2),(1,4),(4,8), крайние подписи 0 и 4 мин, 0 и 10 см. Ломаная состоит из двух отрезков. При замене входа на `[(2,5)]` виден один маркер и нет ненулевого отрезка.

```
import matplotlib
matplotlib.use("Agg")
import matplotlib.pyplot as plt

def chart(rows, path="chart.png", connect=True):
    if len(rows) > 20:
        raise ValueError("too many rows")
    for i, (t, h) in enumerate(rows):
        if not (0 <= t <= 4 and 0 <= h <= 10):
            raise ValueError("range")
        if i and t <= rows[i-1][0]:
            raise ValueError("time order")
    fig, ax = plt.subplots(figsize=(6,4))
    ax.set(xlim=(0,4), ylim=(0,10),
           xlabel="t, min", ylabel="h, cm")
    ax.set_xticks(range(5))
    ax.set_yticks(range(0,11,2))
    if rows:
        x, y = zip(*rows)
        ax.plot(x, y, "o-" if connect else "o")
    else:
        ax.text(2, 5, "No data", ha="center")
    ax.grid(True)
    fig.tight_layout()
    fig.savefig(path, dpi=150)
    plt.close(fig)

chart([(0,2), (1,4), (4,8)])
```



Маркеры — учебные измерения; отрезки соединяют их без добавления наблюдений.

03. Практика 1. Проверяем материал

Проверяем рисунок по координатам

14–25 минут. Сохраните четыре рисунка с разными именами, передав второй аргумент `chart`. Все данные учебные.

Вход	Что должно быть видно
<code>[(0,2),(1,4),(4,8)]</code>	3 маркера, 2 отрезка
<code>[]</code>	No data, нет маркеров
<code>[(2,5)]</code>	Один маркер
<code>[(0,10),(4,0)]</code>	Два крайних маркера, спад

2 минуты — прогноз; 4 — запуск; 3 — проверка числа маркеров, координат, границ и единиц; 2 — исправление намеренной ошибки: вместо времён передайте индексы записей. Форма: имя PNG / четыре проверки / вывод. Критерии: четыре верных рисунка и объяснение искажения интервалов.

04. Практика 2. Перенос способа

Выбираем вид для дискретных наблюдений

25–33 минуты. Серия `[(0,3),(2,7),(4,6)]` описывает три отдельные учебные проверки. За 2 минуты выберите вид, за 4 сохраните `points.png` без отрезков и `line.png` с ними, за 2 запишите различие смысла. Критерии: три одинаковых маркера, разные способы соединения, отсутствие вывода об измерении в минуту 1.

05. Обсуждение и границы вывода

33–40 минут. Один автор обрезал вертикальную шкалу до 6...8, другой оставил 0...10. За 2 минуты опишите визуальный эффект, за 3 оцените сравнимость, за 2 предложите честную подпись и правило для серии рисунков.

06. Индивидуальная проверка

40–45 минут. Для `[(0,1),(4,9)]` назовите число маркеров, наклон, значение на соединяющем отрезке при $t=2$ и статус этого значения. Четыре балла.



Часть 2. Ответы и материалы учителя

Практика 1: разбор и критерии

Индексы [0,1,2] сдвигают последнюю точку основной серии с 4 к 2 минутам. На правильном рисунке последний интервал вдвое длиннее первого. Для [(0,10),(4,0)] наклон -2.5 см/мин. У пустоты нет точки, у единственного наблюдения нет линии между разными точками.

Практика 2: разбор и критерии

Вызовы `chart([(0,3),(2,7),(4,6)], "points.png", False)` и `chart([(0,3),(2,7),(4,6)], "line.png", True)`. В первом три маркера, во втором те же маркеры и два отрезка. Значение 5 в минуту 1 лежит на отрезке, но оно не измерено; это линейная интерполяция.

Обсуждение: возможный вывод

Обрезанная шкала сильнее подчёркивает небольшие различия и скрывает точки ниже 6. Для наших входов сохраняем 0...10, иначе данные теряются. При иной аналитической цели увеличенный фрагмент допустим с явно указанными границами и полным графиком для контекста.

Индивидуальная проверка: ответ и критерии

2 маркера; $(9-1)/4=2$ см/мин; на отрезке $h=5$; это интерполированное значение, не третье наблюдение.

Учителю: подготовка и план на 45 минут

Подготовьте компьютер, редактор, папку с правом записи, распечатку ученической части PDF, бумагу и ручку. Python 3.11+ и IDLE или редактор; запуск `python lesson.py` (либо `python3 lesson.py`). Заранее установите Matplotlib командой `python -m pip install matplotlib`. Код проверен с Matplotlib 3.10.8; графическое окно не требуется, используется Agg. PNG открывайте обычным просмотрщиком. Установку и проверку запуска учитель выполняет заранее. Все программы, данные и ожидаемые ответы даны; регистрация детей и интернет на занятии не нужны.

Уровень — начинающий программист, 9 класс. Предпосылки: оси и пары координат из урока 7, списки и функции. В паре автор прогнозирует и вводит код, проверяющий ведёт журнал, затем роли меняются. Без компьютера выполните трассировку полного кода на бумаге и сопоставьте с готовыми результатами; учитель демонстрирует подготовленный запуск. Бумажный вариант не считается фактическим запуском ученика. Перед практикой учитель выдаёт готовый файл разобранного примера, чтобы дети тратили время на изменение и проверку, а не длинное перепечатывание. Матрица программы предназначена учителю.

Минуты	Работа	Результат
0–5	Проблема	Вопрос и первое решение
5–14	Понятия и пример	Способ проверки
14–25	Практика 1	Готовый продукт и проверка
25–33	Практика 2	Перенос способа
33–40	Обсуждение	Аргумент и ограничение
40–45	Индивидуальная проверка	Самостоятельный ответ



07. Соответствие исходной программе

Пункт программы	Действие ученика	Проверяемый результат
Библиотеки рисования графиков (марафон)	Проверяем рисунок по координатам	Построить PNG по проверенным парам, различить маркеры и соединяющую линию, проверить крайние и пустые серии.
Структурирование решения и отладка	Выбираем вид для дискретных наблюдений	Исходник, журнал обычных и граничных тестов; индивидуальный ответ

Источники и права на изображения

Исходная программа курса

<https://schoolotzyv.ru/blog/python-cpp-9-klass-programma>

Matplotlib: линии и маркеры

https://matplotlib.org/stable/api/_as_gen/matplotlib.axes.Axes.plot.html

Тексты, учебные данные, реплики, программы и схемы подготовлены для уроков портала «Школы России». Числовые серии — учебные модели, не результаты проведённых классом опытов. Схемы авторские, сторонние фотографии не используются. Методические ссылки приведены для учителя; ученик получает все материалы здесь. Проверка первичных источников: 18.10.2026.