



**УЧЕБЕН ПРЕДМЕТ
РАЗРАБОТКА И АНАЛИЗ НА АЛГОРИТМИ**

РАЗРАБОТКА НА ПРОЕКТ

**ТЕМА
„Как компютъра чертае линии “**

РАЗРАБОТИЛ: Иван Тодоров Николов 11а клас №15

ПЛОВДИВ

2023/2024 учебна годин

С Ъ Д Ъ Р Ж А Н И Е

Увод	3
Описание на проблема	4
Алгоритъм за решение	5
Блок схема	6
Решение на проблема	7
Заключение	8
ИЗТОЧНИЦИ	9

Увод

Процесът на чертане на линии от компютъра е възможен благодарение на софтуерни програми и хардуерни компоненти, които работят заедно, за да създадат изображения. Графичните карти и процесорите в компютърните системи използват сложни алгоритми, които определят позициите на пикселите върху екрана, за да формират линии, криви и обекти. Чрез изчисления и математически операции компютърът генерира и пресмята координатите на точките, през които да премине линията, вземайки предвид начална и крайна точка, наклона, дебелината и други параметри.

В днешно време, този процес на чертане на линии чрез компютърни системи е от съществено значение в много области. Той е от особено значение в графичния дизайн, архитектурата, инженерството, научните изследвания и дори във видеоигрите. С помощта на тези технологии се създават реалистични изображения, сложни дигитални модели, анимации и визуализации, които се използват в медицината, проектирането на сгради, развлекателната индустрия и много други сфери.

С обширното развитие на компютърната графика и технологиите за визуализация, възможностите за създаване на впечатляващи и функционални изображения продължават да се разширяват, като компютърите все по-успешно създават визуални решения, които представят и обогатяват реалността в различни области.

Описание на проблема

Проблемът, свързан с това как компютърът чертае линии, се основава на необходимостта да се разработи алгоритъм, който може да генерира линии, като свързва начална и крайна точка в двумерното пространство, използвайки различни параметри като наклон, дебелина на линията и други характеристики. Този проблем има приложения във визуалната репрезентация на данни, създаването на изображения, графики и дигитални дизайни. Решението на този проблем е от голямо значение за широк спектър от области, включително графичния дизайн, инженерството, компютърната анимация и дори в научните изследвания.

Как може да бъде полезно на днешно време:

1. **Графичен дизайн:** Графичните дизайнери използват алгоритмите за чертане на линии, за да създават изключителни логота, банери, илюстрации и рекламни материали. Този процес позволява създаването на креативни и професионални изображения.
2. **Архитектура и инженерство:** В архитектурната и инженерната област, компютрите използват алгоритмите за чертане на линии, за да създават точни чертежи и проекти на сгради и инженерни системи.
3. **Видеоигри:** В сферата на развлекателната индустрия, алгоритмите за чертане на линии се използват за рендиране на игрови сцени и обекти, което води до по-реалистични и впечатляващи видеоигри.
4. **Медицински изображения:** В медицинските приложения, алгоритмите за чертане на линии се използват за визуализация на медицински изображения като рентгени и скенери.
5. **Научни изследвания:** В различни научни области, включително физиката и географията, алгоритмите за чертане на линии са от съществено значение при изобразяването на данни и резултатите от изследвания.

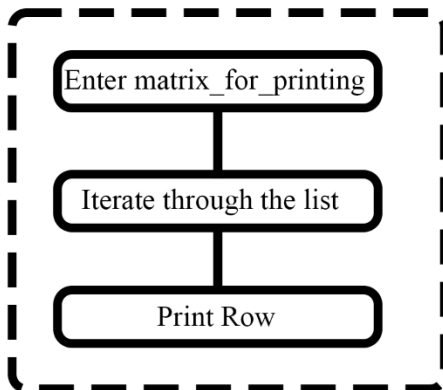
Обобщено, развитието на алгоритми за чертане на линии в програмния език Python и тяхната интеграция в различни приложения играе критична роля в подобряването на визуалната комуникация и в създаването на по-реалистични и функционални изображения и графики в днешното информационно общество.

Алгоритъм за решение

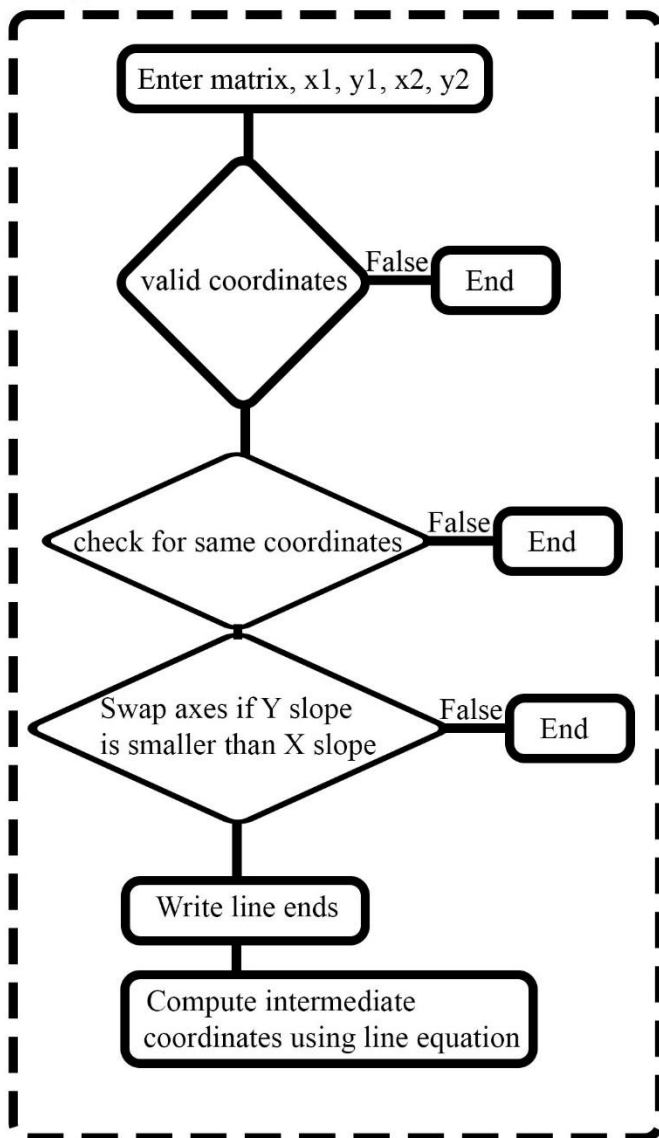
1. Приемане на два параметъра (X, Y) на първата точка
2. Приемане на два параметъра (X, Y) на втората точка
3. Разделяме дойките точки на 4 параметъра
4. Използваме вградената библиотека `num_ru` за да построим координатна система в която всички клетки са със стойност 0
5. Проверяваме дали въведените точки са валидни координати в координатната система
6. Проверяваме дали точките съвпадат ако да прекратяваме програмата и принтираме координатната система
7. Разменяме X и Y в координатната система ако хикса е по малък кат стойности
8. Ако е нужно определяме посоката в която да се чертае линията
9. Генерираме лист от числа, започващ от $(0 + 1)$ и върви до $(x1 - 1)$. Това са x-координатите на точките в отсечката.
10. Смятаме съответните y-координати за всяка точка от x-отсечката, използвайки линейна интерполация. (за повече информация по долу в документа)
11. Записваме линията в координатната система
12. Изпринтираме кординатната система

Блок схема

print_matrix



draw_line



Решение на проблема

```
import numpy as np

ROW = 20
COL = 20

def print_matrix(matrix_for_printing):
    for r in matrix_for_printing:
        print(*r, sep=" ")

def draw_line(mat, x0, y0, x1, y1, inplace=False):
    if not (0 <= x0 < mat.shape[0] and 0 <= x1 < mat.shape[0] and
            0 <= y0 < mat.shape[1] and 0 <= y1 < mat.shape[1]):
        raise ValueError('Invalid coordinates.')

    if not inplace:
        mat = mat.copy()

    if (x0, y0) == (x1, y1):
        mat[x0, y0] = 2
        return mat if not inplace else None

    # Swap axes if Y slope is smaller than X slope
    transpose = abs(x1 - x0) < abs(y1 - y0)
    if transpose:
        mat = mat.T
        x0, y0, x1, y1 = y0, x0, y1, x1
    # Swap line direction to go left-to-right if necessary
    if x0 > x1:
        x0, y0, x1, y1 = x1, y1, x0, y0
    # Write line ends
    mat[x0, y0] = 2
    mat[x1, y1] = 2
    # Compute intermediate coordinates using line equation
    x = np.arange(x0 + 1, x1)
    y = np.round(((y1 - y0) / (x1 - x0)) * (x - x0) + y0).astype(x.dtype)
    # Write intermediate coordinates
    mat[x, y] = 1
    if not inplace:
        # return mat if not transpose else mat.T
        print_matrix(mat)

x, y = input("Първа точка(x y): ").split()
a, b = input("Втора точка(x y): ").split()

first_point_x = int(x)
first_point_y = int(y)
second_point_x = int(a)
second_point_y = int(b)

print(draw_line(np.zeros((ROW, COL), dtype=str), first_point_x,
first_point_y, second_point_x, second_point_y))
```

Заклучение

Заклучението на доклада за това как компютърът чертае линии с помощта на NumPy би подчертало важността на библиотеката NumPy в обработката на данни и визуализацията. Представената информация вероятно би обхванала начина, по който NumPy използва математически операции и алгоритми, за да изобрази линии в компютърната графика. В заключението би се подчертало как NumPy предоставя мощни инструменти за обработка на данни и създаване на графики, които са от съществено значение за различни области като наука, инженерство и компютърни науки. То ще подчертае значението на използването на библиотеки като NumPy за постигане на висококачествени и точни визуализации в програмирането и анализа на данни.

Источници

1. <https://docs.python.org>
2. <https://stackoverflow.com>
3. <https://www.wikipedia.org>
4. <https://chat.openai.com>
5. <https://support.microsoft.com>
6. <https://duckduckgo.com>
7. <https://www.jetbrains.com/pycharm>