

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Вятский государственный университет»
Факультет автоматики и вычислительной техники
Кафедра электронных вычислительных машин

РЕАЛИЗАЦИЯ АЛГОРИТМОВ РАСТРОВОЙ РАЗВЕРТКИ ОКРУЖНОСТЕЙ
Отчет по лабораторной работе №2 дисциплины
«Компьютерная графика»

Выполнил студент группы ИВТ-21 _____/Рзаев А.Э./
Проверил старший преподаватель _____/Вожегов Д.В./

2016 г.

1 Постановка задачи

Написать и отладить программу, реализующую два алгоритма построения окружности: по формуле $Y = \pm \sqrt{r^2 - x^2}$ и Брезенхема. В обоих случаях использовать свойство симметрии окружности (в первом - найдя точки четверти окружности, остальные - отразив симметрично; во втором - свойство симметрии использовать полностью).

2 Словесное описание алгоритмов

Существуют несколько простых способов преобразования окружности в растровую форму. Например, по формуле $X^2 + Y^2 = R^2$ для окружности с центром в начале координат. Чтобы изобразить четверть такой окружности, на каждом шаге следует поменять X от 0 до R на единицу и вычислить Y как $\sqrt{R^2 - X^2}$. Остальные четверти изображают симметрично. Этот метод содержит операции умножения и извлечения корня, потому не эффективен. Кроме того, при X , близких к R , в окружности появляются заметные промежутки, так как при таких X тангенс угла наклона касательной к окружности стремится к бесконечности. Процесс можно улучшить, если вычислять одну восьмую часть окружности, а остальные семь частей отображать симметрично (в предыдущем случае X менять от 0 до $R/\sqrt{2}$). Но необходимый эффект можно получить только при работе с целыми числами.

3 Вывод

Для каждого двух алгоритмов (простой и алгоритм Брезенхема) были написаны соответствующие процедуры, реализующие их. Наиболее эффективным и правильным является, как и в прошлой лабораторной работе, алгоритм Брезенхема, в котором полностью используется свойство симметрии окружности.

Блок-схемы алгоритмов, листинг процедур, реализующих алгоритмы рисования окружностей, и экранные формы программы приведены в приложениях А, Б и В.

Приложение А
(обязательное)
Блок-схемы алгоритмов

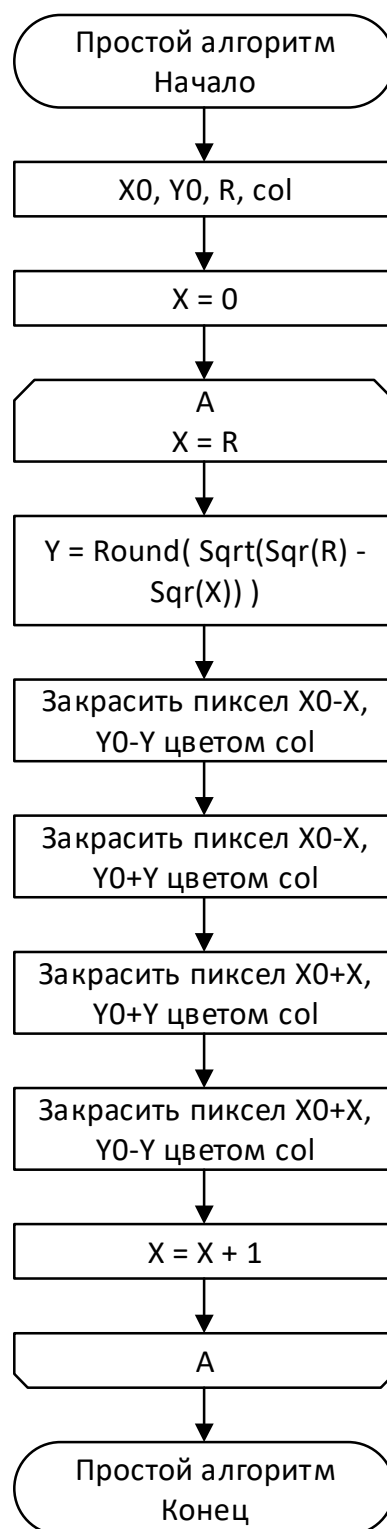


Рисунок А.1 – Схема простого алгоритма развертки окружности



Рисунок А.2 – Схема восьмисвязного алгоритма Брезенхема

Приложение Б
(обязательное)
Листинг процедур

```
def circle_simple_general(self):
    # простой алгоритм
    x0, y0 = self.x0_var.get(), self.y0_var.get()
    r = self.radius_var.get()
    for x in range(r + 1):
        y = round((r * r - x * x) ** 0.5)

        self.draw_pixel(x0 - x, y0 - y)
        self.draw_pixel(x0 - x, y0 + y)
        self.draw_pixel(x0 + x, y0 + y)
        self.draw_pixel(x0 + x, y0 - y)

def circle_brezenhem_general(self):
    # алгоритм Брезенхема
    x0, y0 = self.x0_var.get(), self.y0_var.get()
    r = self.radius_var.get()

    x, y = 0, r
    e = 3 - 2 * r

    while x <= y:
        self.draw_pixel(x0 + x, y0 + y)
        self.draw_pixel(x0 + y, y0 + x)
        self.draw_pixel(x0 + y, y0 - x)
        self.draw_pixel(x0 + x, y0 - y)
        self.draw_pixel(x0 - x, y0 - y)
        self.draw_pixel(x0 - y, y0 - x)
        self.draw_pixel(x0 - y, y0 + x)
        self.draw_pixel(x0 - x, y0 + y)

        if e >= 0:
            e += 4 * (x - y) + 10
            y -= 1
        else:
            e += 4 * x + 6

        x += 1
```

Приложение В
(обязательное)
Экранные формы программы

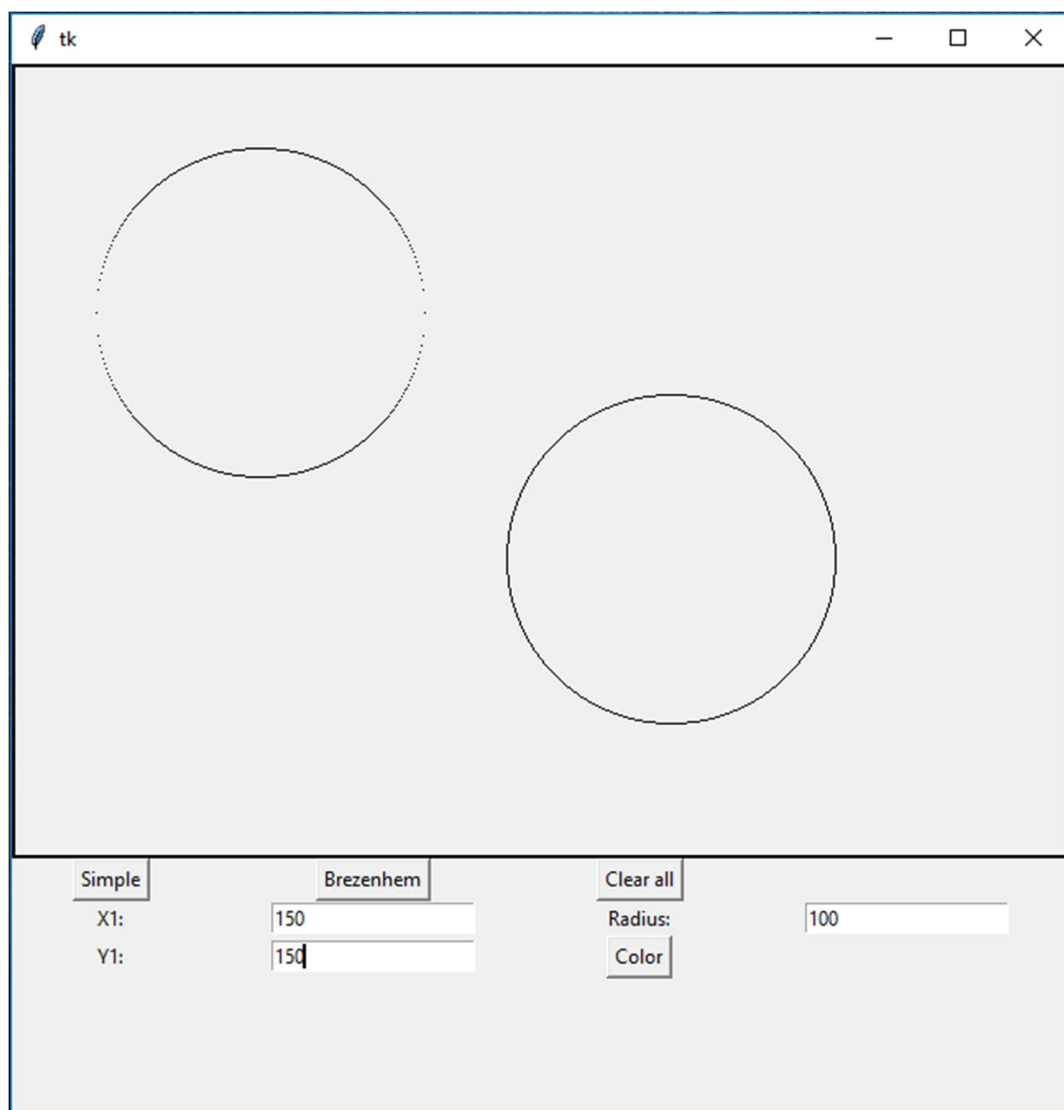


Рисунок В.1 – Развертка окружностей простым алгоритмом и алгоритмом Брезенхема