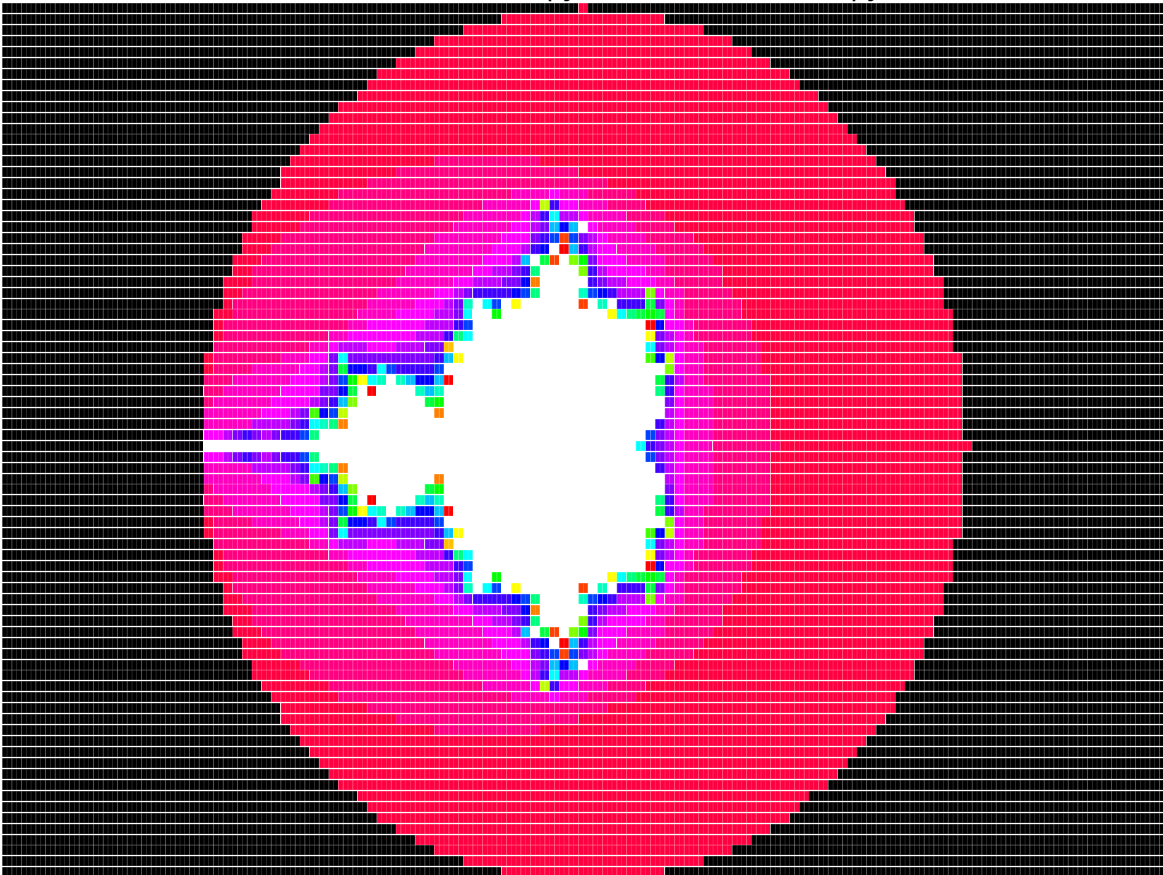
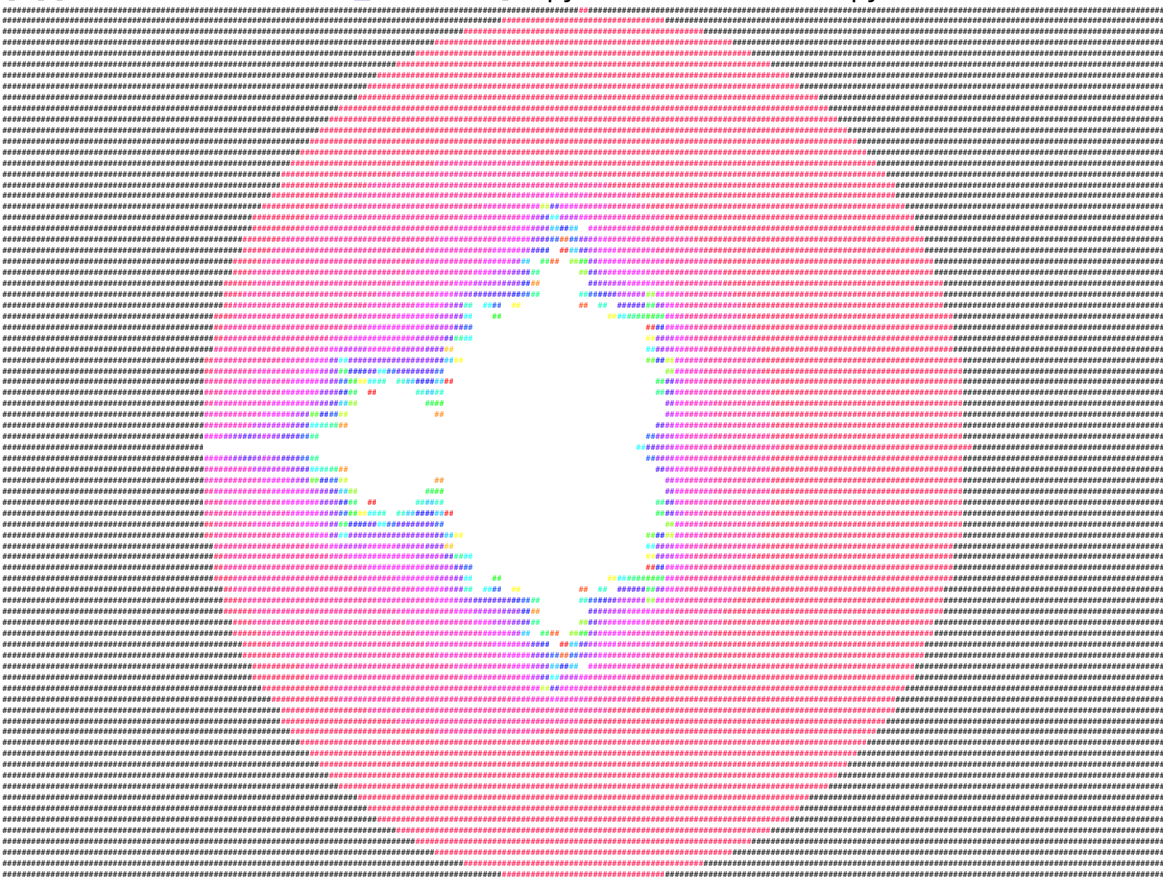


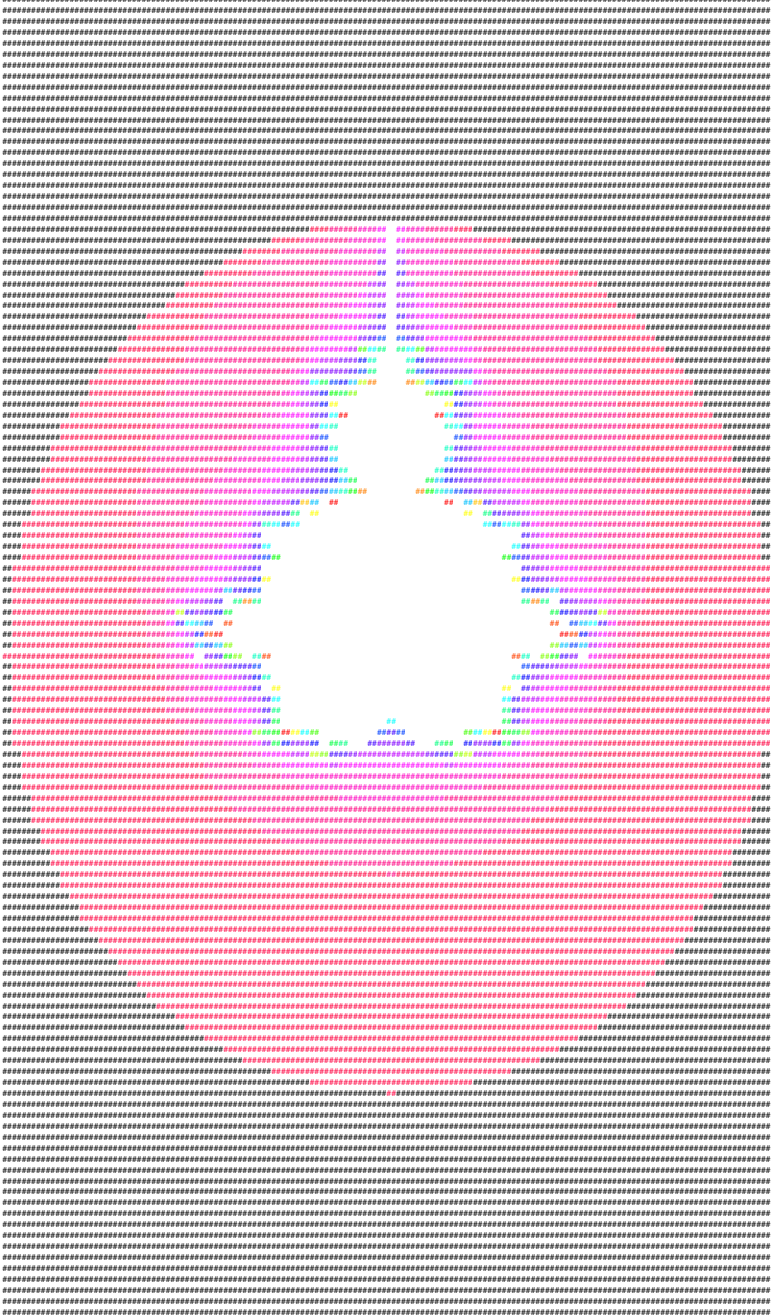
```
[0][buddha@tuxedo math_research]$ python3 mandelbrot.py -3 -2 3 2 0.05 2 2 " " 0
```



```
[0][buddha@tuxedo math_research]$ python3 mandelbrot.py -3 -2 3 2 0.05 2 2 "#" 0
```



```
[0][buddha@tuxedo math_research]$ python3 mandelbrot.py -3 -2 3 2 0.05 2 2 "#" 1
```



Programcode:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Created on Sat Feb  8 16:04:18 2025

@author: buddha
"""

import sys
import math

STARTX = -2
STARTY = -1.5
ENDX = 1
ENDY = 1.5
STEP = 0.05
SIZE = 2
BORDER = 2
CHARACTER = "*"
FOREGROUND = True
UPRIGHT = True

if len(sys.argv) > 1:
    if len(sys.argv) == 5:
        STARTX = float(sys.argv[1])
        STARTY = float(sys.argv[2])
        ENDX = float(sys.argv[3])
        ENDY = float(sys.argv[4])
    elif len(sys.argv) == 9:
        STARTX = float(sys.argv[1])
        STARTY = float(sys.argv[2])
        ENDX = float(sys.argv[3])
        ENDY = float(sys.argv[4])
        STEP = float(sys.argv[5])
        SIZE = int(sys.argv[6])
        BORDER = float(sys.argv[7])
        CHARACTER = sys.argv[8]
        if CHARACTER == " ":
            CHARACTER = "\u2588"
    elif len(sys.argv) == 10:
        STARTX = float(sys.argv[1])
        STARTY = float(sys.argv[2])
        ENDX = float(sys.argv[3])
        ENDY = float(sys.argv[4])
        STEP = float(sys.argv[5])
        SIZE = int(sys.argv[6])
        BORDER = float(sys.argv[7])
        CHARACTER = sys.argv[8]
        UPRIGHT = True if sys.argv[9] == "1" else False
    else:
```

```

        print(f"Usage: python3 {sys.argv[0]} [<startx> <starty> <endx> <endy>
[<step>"]
        " <pixelsize> <colorborder> <character> <upright>]]")
        sys.exit(1)

colors = [[255, 255, 255], [255, 0, 0], [255, 64, 0], [255, 128, 0],
          [255, 192, 0], [255, 255, 0], [192, 255, 0], [128, 255, 0],
          [64, 255, 0], [0, 255, 0], [0, 255, 64], [0, 255, 128],
          [0, 255, 192], [0, 255, 255], [0, 192, 255], [0, 64, 255],
          [0, 0, 255], [64, 0, 255], [128, 0, 255], [192, 0, 255],
          [255, 0, 255], [255, 0, 192], [255, 0, 128], [255, 0, 64],
          [0, 0, 0]]

def colorize(text, r, g, b, fg=True):
    charcode = "38" if fg else "48"
    return f"\033[{charcode};2;{r};{g};{b}m{text}\033[0m"

def frange(start, end, step):
    while start < end:
        yield start
        start += step

def get_value(real, imag, cimag, creal, border, depht=(len(colors) - 1)):
    if depht:
        xposnew = real ** 2 + creal - imag ** 2
        yposnew = 2 * real * imag + cimag
        if math.sqrt(xposnew ** 2 + yposnew ** 2) > border:
            return depht
        return get_value(xposnew, yposnew, cimag, creal, border, depht-1)
    else:
        return 0

if not UPRIGHT:
    for imag in frange(STARTY, ENDY, STEP):
        for real in frange(STARTX, ENDX, STEP):
            value = get_value(0, 0, imag, real, BORDER)
            print(colorize(CHARACTER*SIZE, *colors[value]), end="")
        print()
else:
    for real in frange(STARTX, ENDX, STEP):
        for imag in frange(STARTY, ENDY, STEP):
            value = get_value(0, 0, imag, real, BORDER)
            print(colorize(CHARACTER*SIZE, *colors[value]), end="")
        print()

```