(MAC 0210)
Exercício Programa 1 - Equações não lineares em uma variável.

Lucas Martins Próspero - 15471925

26 de maio de 2025

Conteúdo

1 Parte 1: Método de ponto fixo

1.1 Escolha de $g(x)$

Na implementação do método de ponto fixo, testou-se as seguintes funções auxiliares $g(x)$:

$$g(x) = f(x) + x, \tag{1}$$

$$g(x) = 2 \cdot f(x) + x, \tag{2}$$

$$g(x) = x - \frac{f(x)}{f'(x)}. \tag{3}$$

Essas escolhas, que são convencionais, foram baseadas no tópico que aborda esse método numérico no livro *A First Course in Numerical Methods - U.M. Ascher, Chen Greif*.

1.2 Amostragem de $f(x)$

A amostragem de $f(x)$, apesar de não ter sido requisitada no enunciado do EP, além das funções auxiliares e suas respectivas derivadas (o que será útil na interpretação dos resultados), foi feita em *Python*, e sua análise inicial permite uma busca mais concentrada das raízes (Figura 1.1).

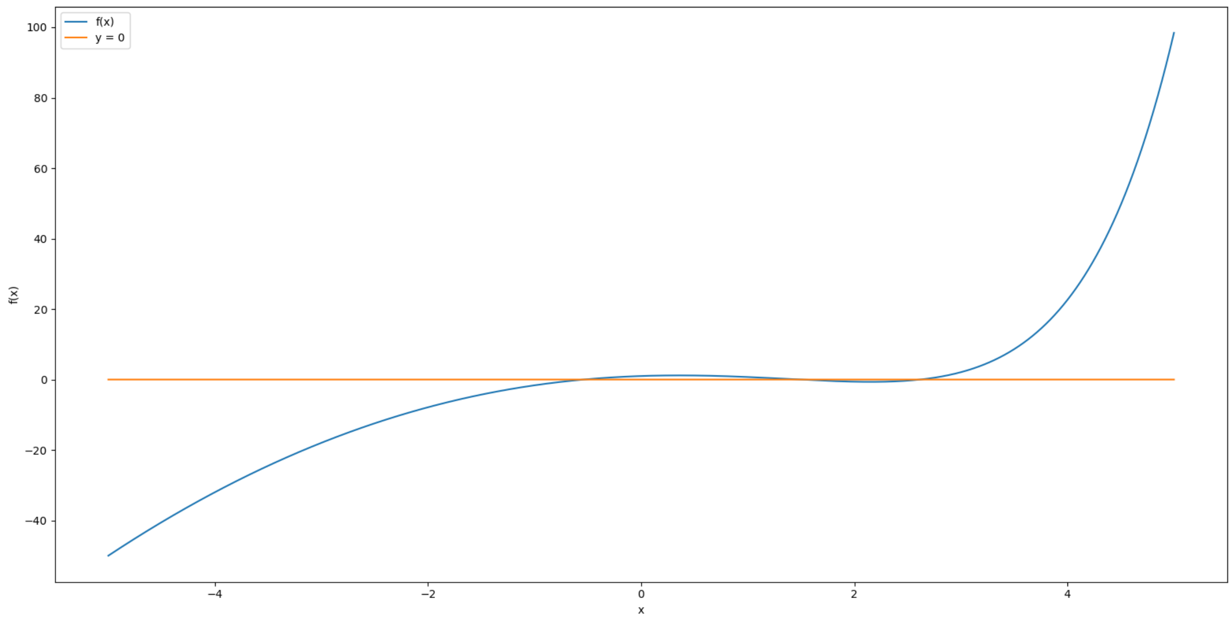


Figura 1.1

O gráfico, apesar de ter sido amostrado no intervalo $[-5, 5]$, revela uma boa intuição de que todas as raízes estão ali presentes, já que em $(-\infty, -5)$ a função se comporta de maneira estritamente decrescente (e^x assume valores positivos muito pequenos conforme x diminui), enquanto que em $(5, +\infty)$ se comporta de maneira estritamente crescente (e^x supera $-2x^2$ rapidamente com o incremento de x). Uma vez feita a amostragem, e ampliando-se a visão do gráfico, é plausível que a busca das raízes seja feita nos intervalos $[-0.6, -0.4]$, $[1.4, 1.6]$ e $[2.6, 2.7]$.

1.3 Análise de $g(x) = f(x) + x$

Como a condição de parada da implementação é $|x_{k+1} - x_k| \leq e_{abs(x)}$, foca-se em analisar o critério de convergência do método ($|g'(x)| < 1$), o que se repetirá nas demais análises.

Intervalo $[-0.6, -0.4]$

Em $[-0.6, -0.4]$, temos $|g'(x)| > 1$, o que dispensa a aplicação do método para aproximar a raiz nesse intervalo (Figura 1.2).

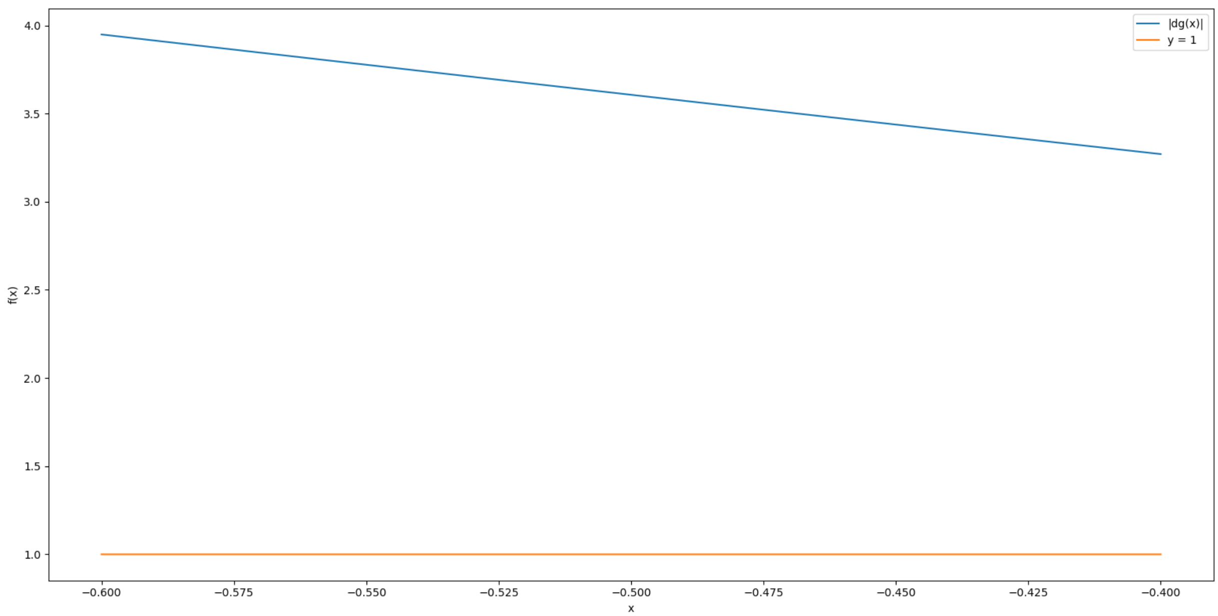
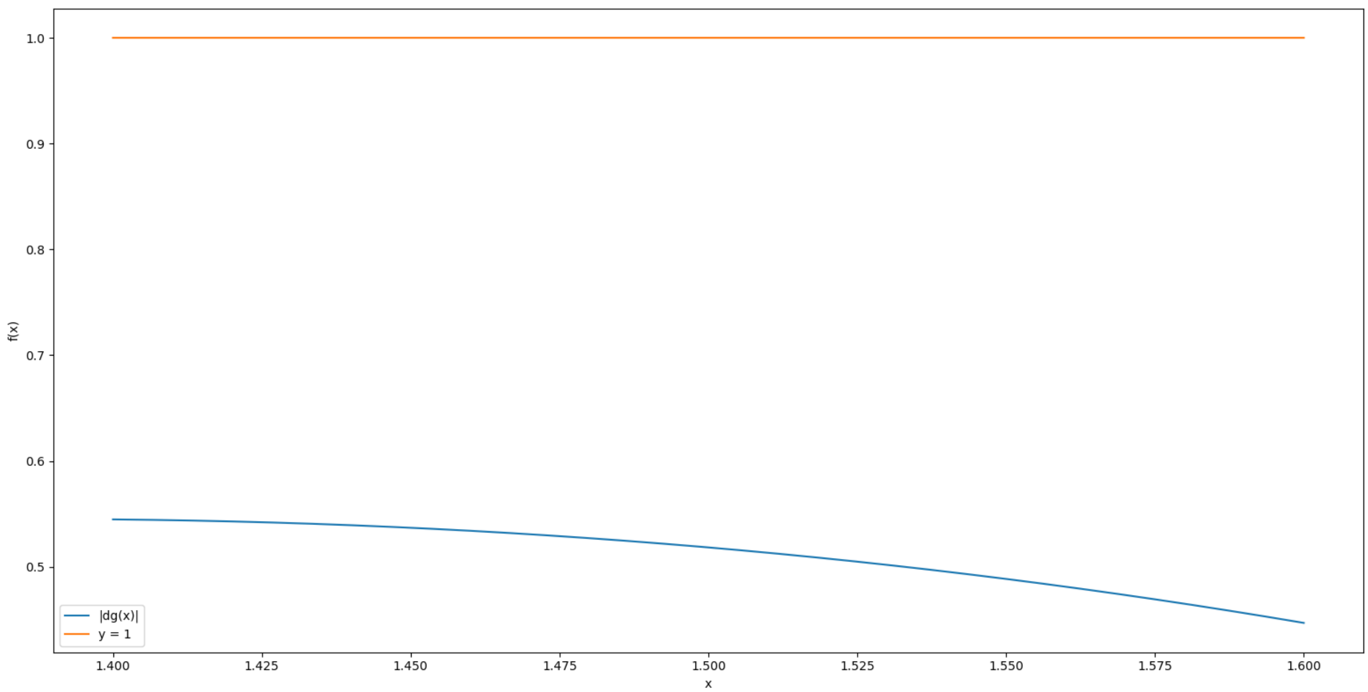


Figura 1.2

Intervalo $[1.4, 1.6]$

Em $[1.4, 1.6]$, temos $|g'(x)| < 1$, o que garante a convergência do método (Figura 1.3). Além disso, visualiza-se a existência de um único ponto fixo nesse intervalo, candidato a raiz de $f(x)$ (Figura 1.4).



1.0
0.9
0.8
0.7
0.6
0.5
f(x)
|dg(x)|
y = 1
1.400
1.425
1.450
1.475
1.500
1.525
1.550
1.575
1.600
x

Figura 1.3

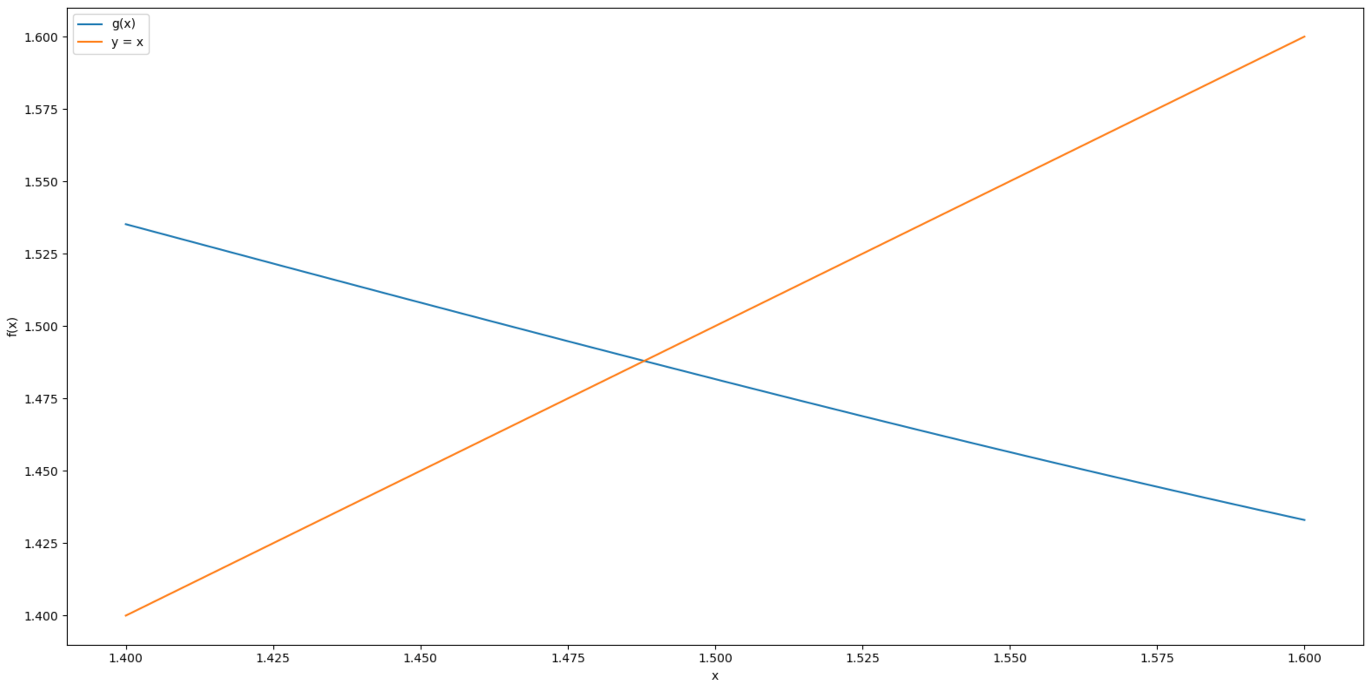


Figura 1.4

Intervalo $[2.6, 2.7]$

Em $[2.6, 2.7]$, temos $|g'(x)| > 1$, o que dispensa a aplicação do método para aproximar a raiz nesse intervalo (Figura 1.5).

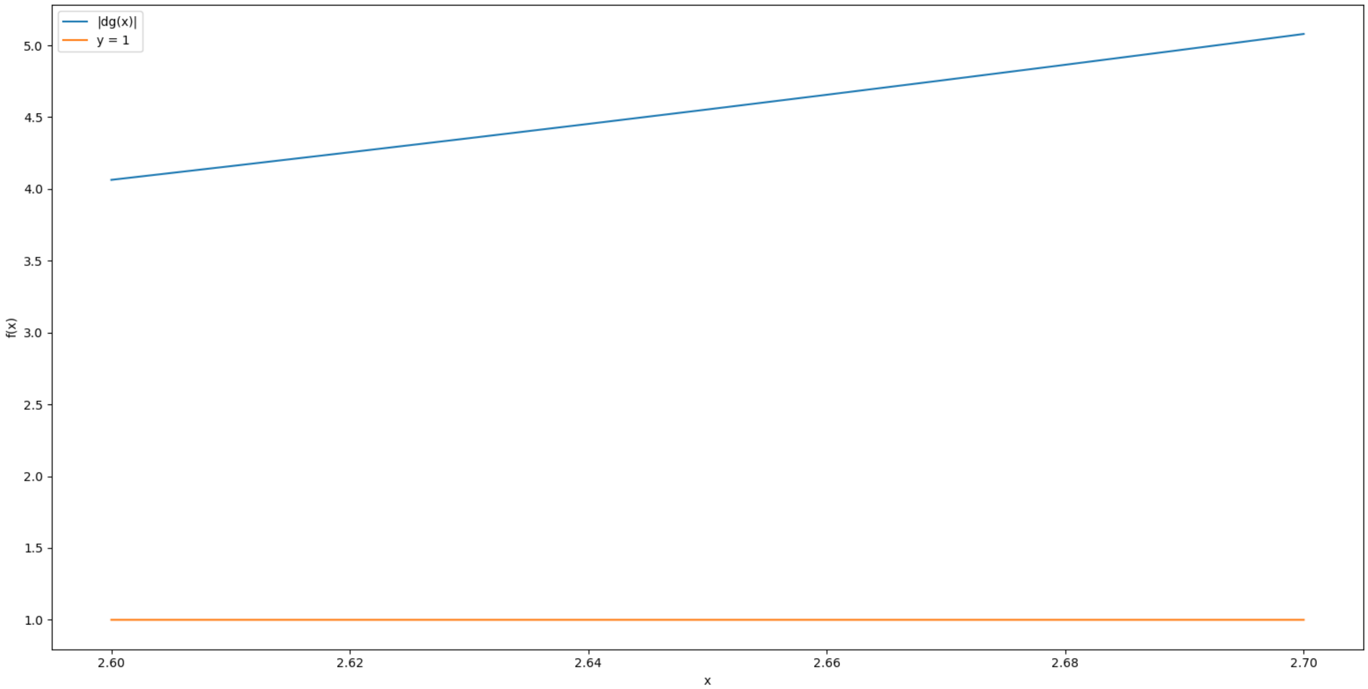


Figura 1.5

1.4 Análise de $g(x) = 2 \cdot f(x) + x$

Intervalo $[-0.6, -0.4]$

Em $[-0.6, -0.4]$, temos $|g'(x)| > 1$, o que dispensa a aplicação do método para aproximar a raiz nesse intervalo (Figura 1.6).

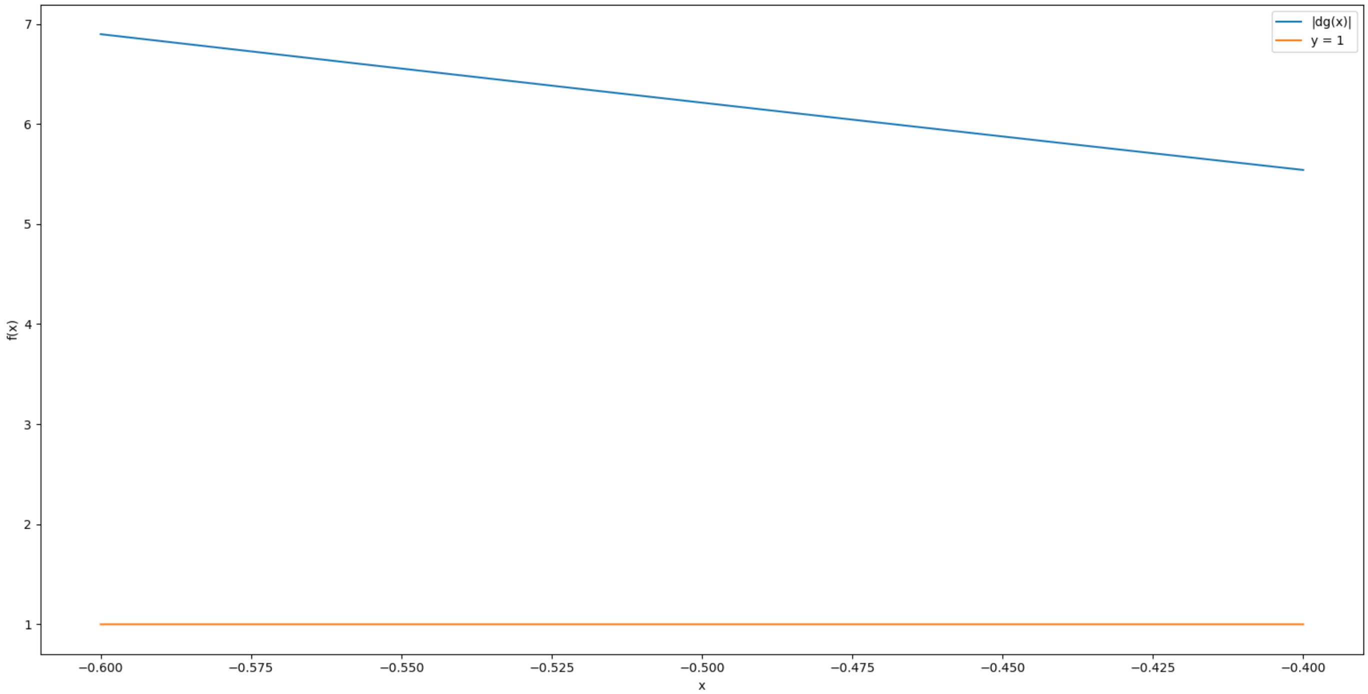


Figura 1.6

Intervalo $[1.4, 1.6]$

Em $[1.4, 1.6]$, temos $|g'(x)| > 1$, o que dispensa a aplicação do método para aproximar a raiz nesse intervalo (Figura 1.7).

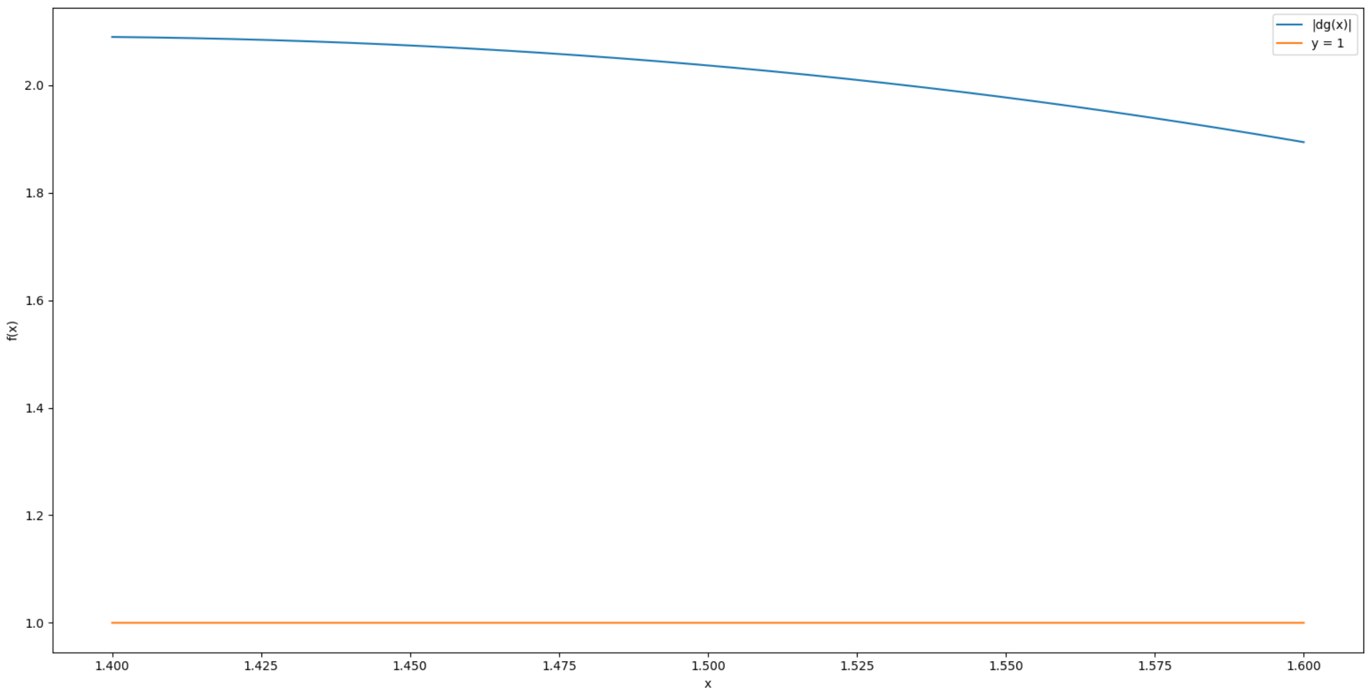


Figura 1.7

Intervalo $[2.6, 2.7]$

Em $[2.6, 2.7]$, temos $|g'(x)| > 1$, o que dispensa a aplicação do método para aproximar a raiz nesse intervalo (Figura 1.8).

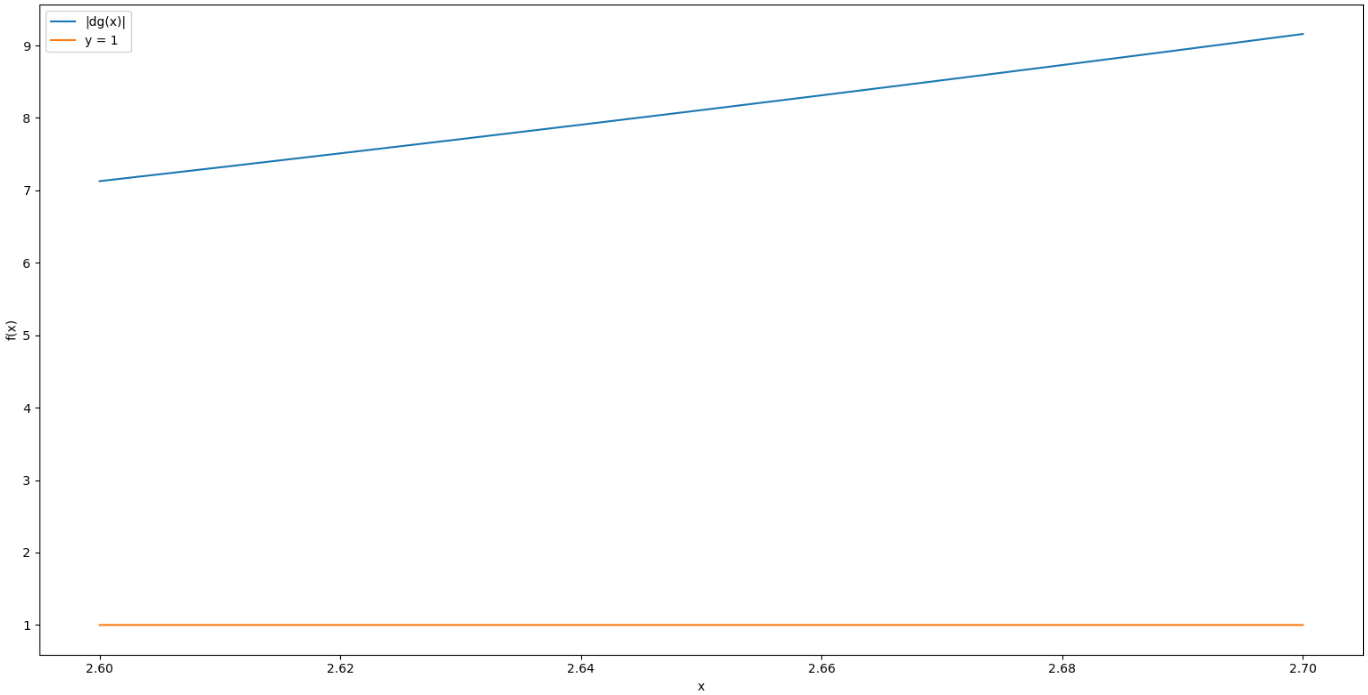
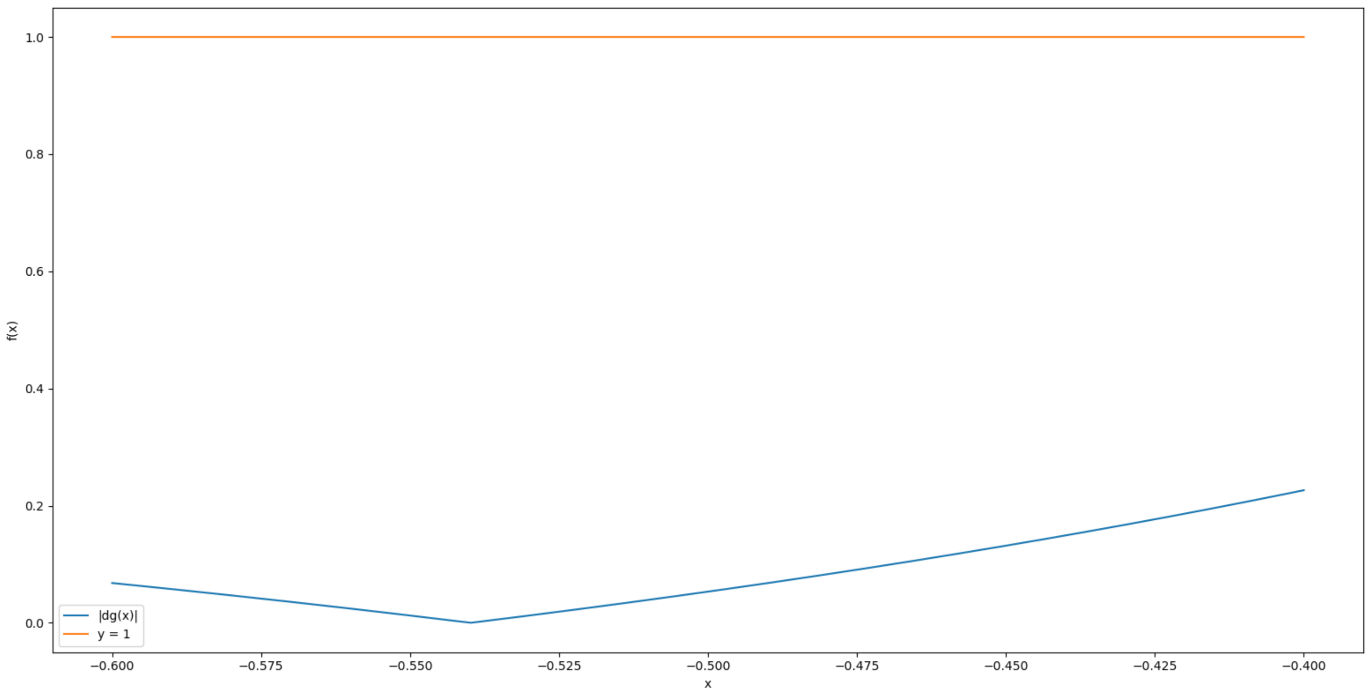


Figura 1.8

1.5 Análise de $g(x) = x - \dfrac{f(x)}{f'(x)}$

Intervalo $[-0.6, -0.4]$

Em $[-0.4, -0.6]$, temos $|g'(x)| < 1$, o que garante a convergência do método (Figura 1.9). Além disso, visualiza-se a existência de um único ponto fixo nesse intervalo, candidato a raiz de $f(x)$ (Figura 1.10).



1.0
0.8
0.6
0.4
0.2
0.0
f(x)
−0.600
−0.575
−0.550
−0.525
−0.500
−0.475
−0.450
−0.425
−0.400
x
|dg(x)|
y = 1

Figura 1.9

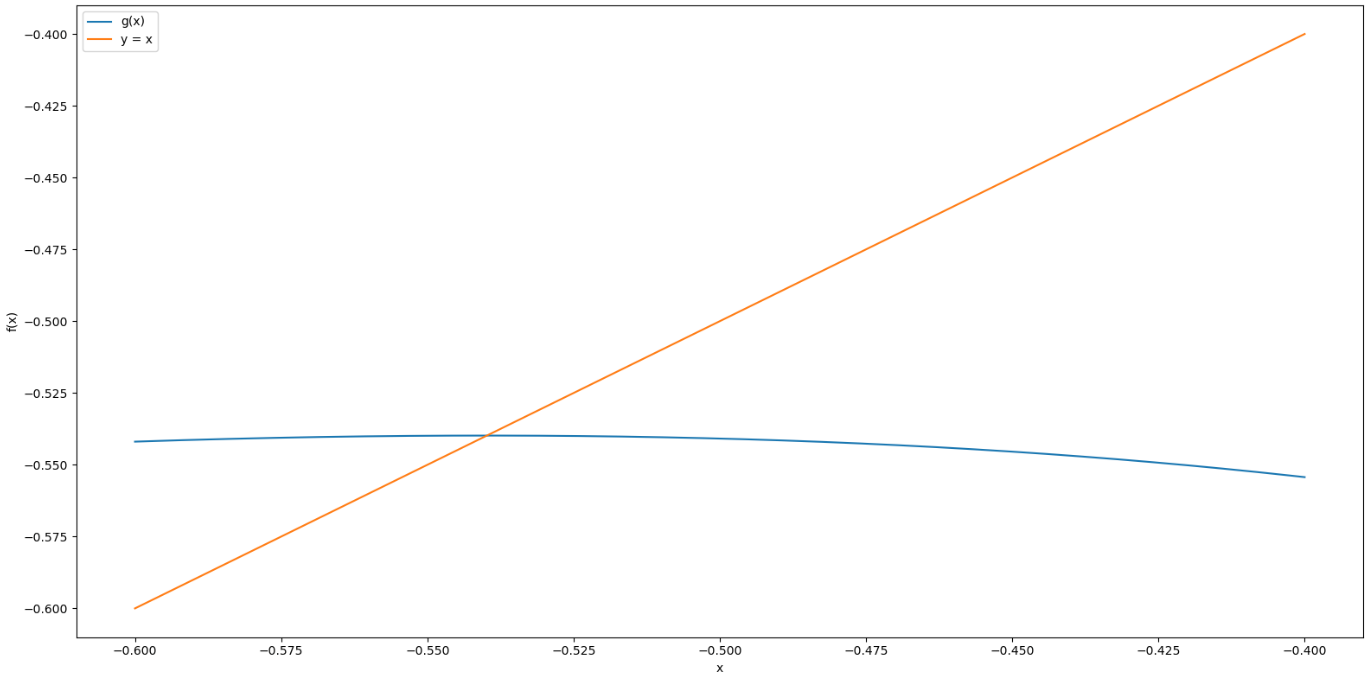


Figura 1.10

Intervalo $[1.4, 1.6]$

Em $[1.4, 1.6]$, temos $|g'(x)| < 1$, o que garante a convergência do método (Figura 1.11). Além disso, visualiza-se a existência de um único ponto fixo nesse intervalo, candidato a raiz de $f(x)$ (Figura 1.12).

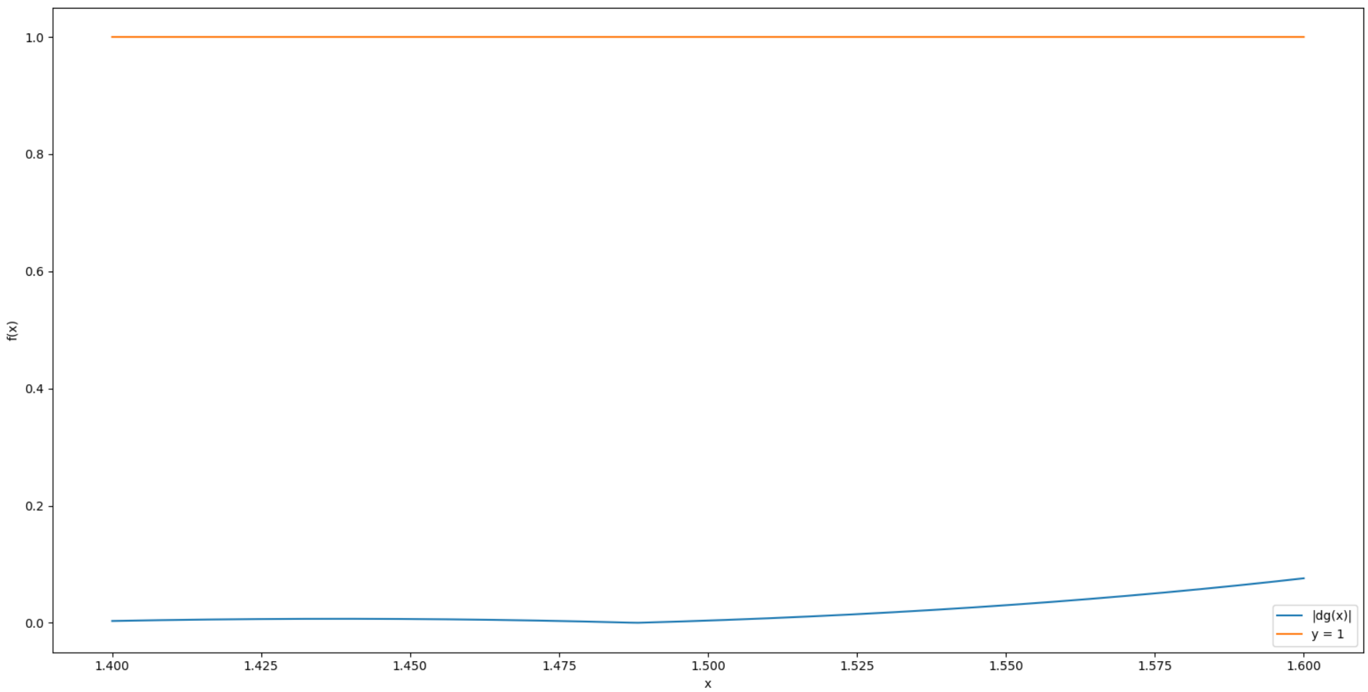

1.0
0.8
0.6
0.4
0.2
0.0
f(x)
1.400
1.425
1.450
1.475
1.500
1.525
1.550
1.575
1.600
x
|dg(x)|
y = 1

Figura 1.11

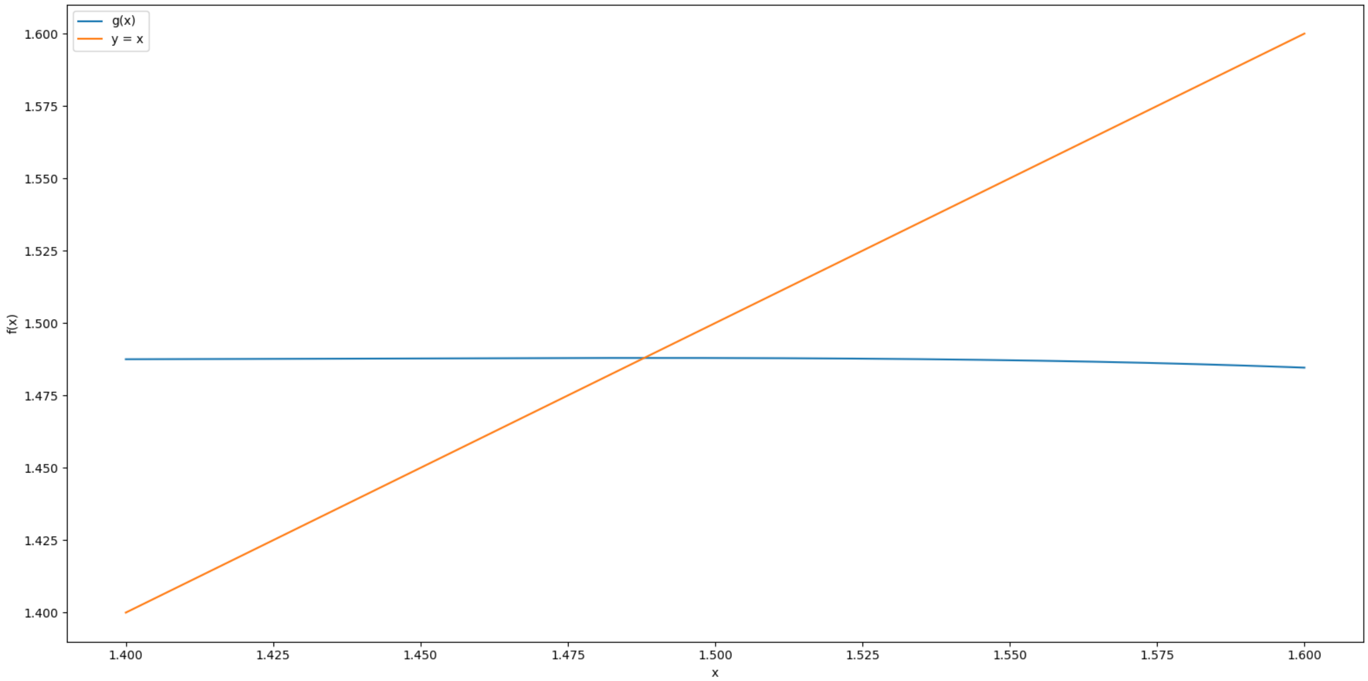


Figura 1.12

Intervalo $[2.6, 2.7]$

Em $[2.6, 2.7]$, temos $|g'(x)| < 1$, o que garante a convergência do método (Figura 1.13). Além disso, visualiza-se a existência de um único ponto fixo nesse intervalo, candidato a raiz de $f(x)$ (Figura 1.14).

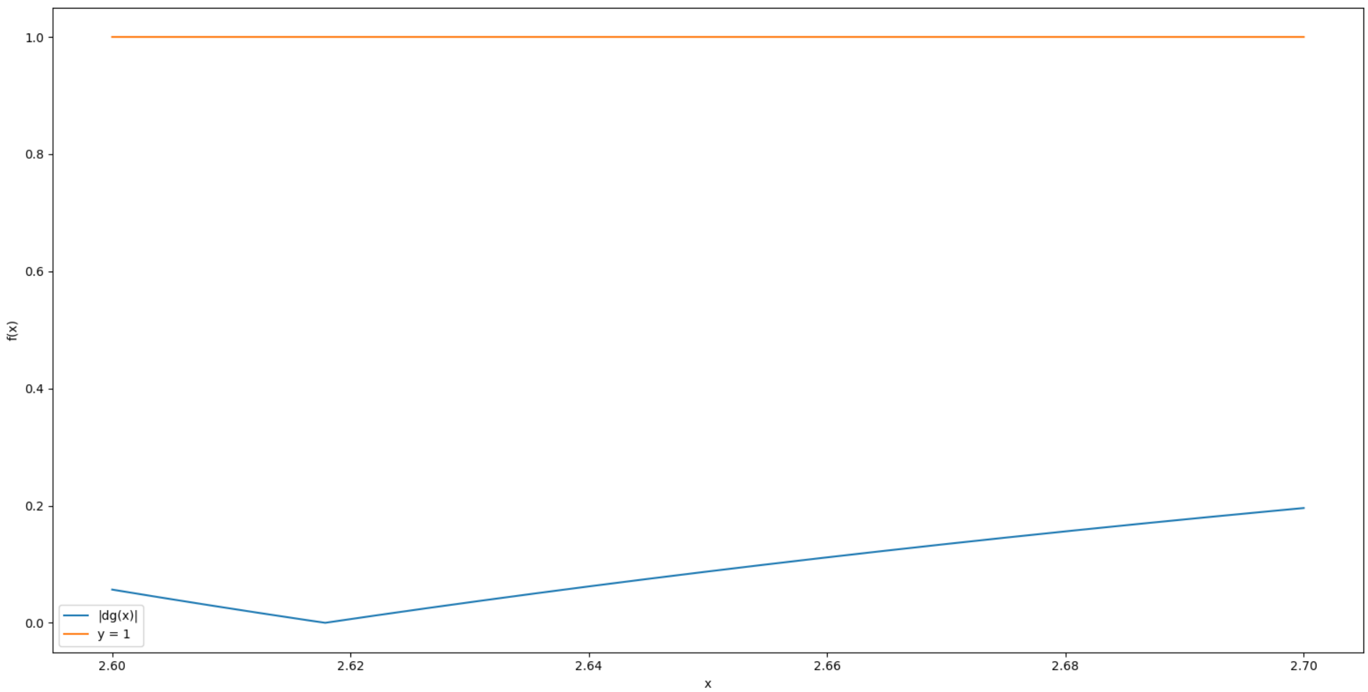


Figura 1.13

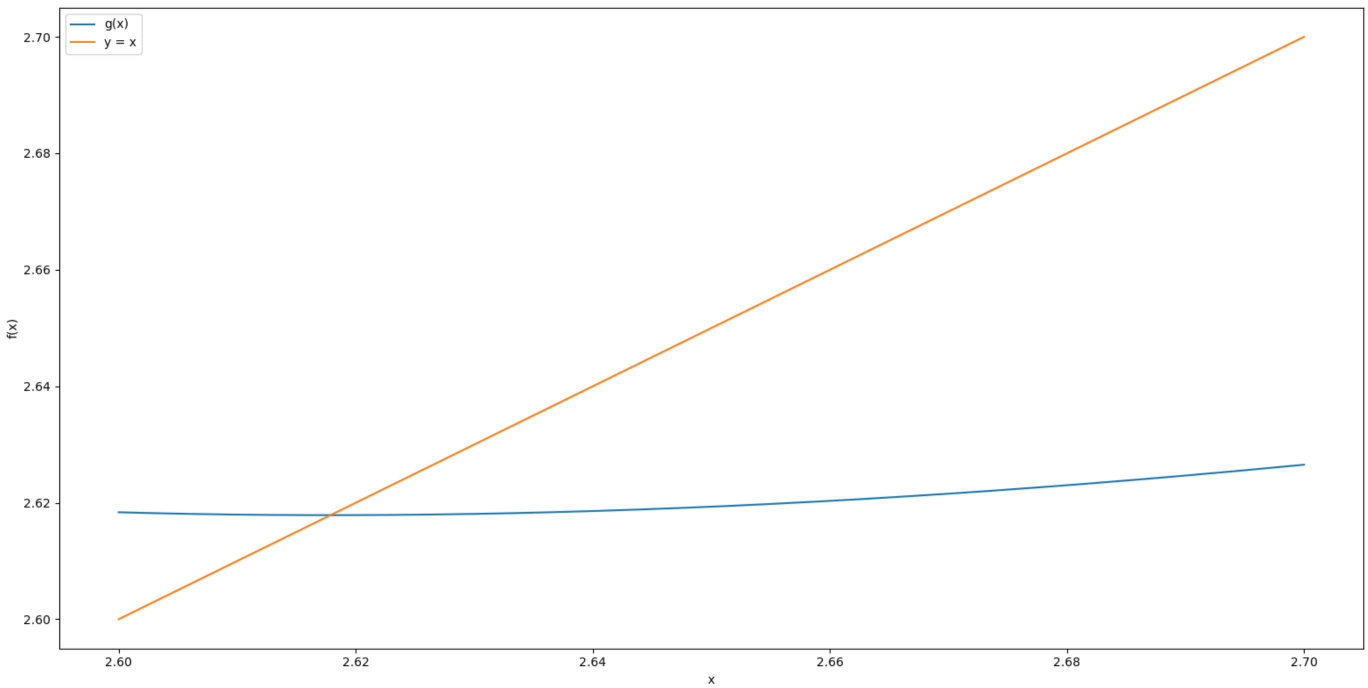


Figura 1.14

1.6 Implementação

O código referente à implementação do método do ponto fixo já se encontra comentado e explicado (ver **(EP1_1.c)**). Como citado anteriormente, escolheu-se a função $g(x) = x - \frac{f(x)}{f'(x)}$ para iterar sobre $x_{k+1} = g(x_k)$. Além disso, os critérios de parada são controlados pelo parâmetro **eabs** (medida do erro absoluto a cada iteração) e pela variável **maxit** (escolheu-se 100 iterações de maneira arbitrária para evitar oscilações). A execução do programa aplica o método do ponto fixo nos pontos iniciais -0.5, 1.47 e 2.6, escolhidos estrategicamente por amostragem prévia (ver **graph.py**), e mostra a tendência de convergência (ver Tabela 1).

x_0	**Erros**
-0.5	$4.087067202362 \times 10^{-2}$
	$1.034727943491 \times 10^{-3}$
	$6.671770352851 \times 10^{-7}$
	$2.773337115514 \times 10^{-13}$
1.47	$1.792250514673 \times 10^{-2}$
	$3.956013168338 \times 10^{-5}$
	$2.197606541188 \times 10^{-10}$
2.6	$1.836383018246 \times 10^{-2}$
	$4.968466004742 \times 10^{-4}$
	$3.705149653754 \times 10^{-7}$
	$2.065014825803 \times 10^{-13}$

Tabela 1: Erros por iteração para diferentes valores de x_0 no método do ponto fixo

A tabela mostra, na prática, não apenas a tendência de convergência do método, mas também uma taxa quadrática, o que não é à toa, já que a escolha $g(x) = x - \frac{f(x)}{f'(x)}$ é similar ao método de Newton ($x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$), o qual possui convergência quadrática.

2 Parte 2: Método de Newton

2.1 Escolha de f(x)

As funções escolhidas a serem aplicadas no método, e suas respectivas derivadas, são:

- $f_1(x) = x^4 + 1$, $f_1'(x) = 4x^3$,
- $f_2(x) = x^5 - 1$, $f_2'(x) = 5x^4$.

As raízes de $f_1(x)$ e $f_2(x)$ (importantes para o código) são respectivamente:

- $z_0 = \text{cis}\left(\frac{\pi}{4}\right)$, $\quad z_1 = \text{cis}\left(\frac{3\pi}{4}\right)$, $\quad z_2 = \text{cis}\left(\frac{5\pi}{4}\right)$, $\quad z_3 = \text{cis}\left(\frac{7\pi}{4}\right)$,
- $r_0 = \text{cis}(0)$, $\quad r_1 = \text{cis}(\frac{2\pi}{5})$, $\quad r_2 = \text{cis}(\frac{4\pi}{5})$, $\quad r_3 = \text{cis}(\frac{6\pi}{5})$, $\quad r_4 = \text{cis}(\frac{8\pi}{5})$.

2.2 Bacias de convergência

Para a função $f_1(x) = x^4 + 1$, obteve-se 4 zonas fractais no plano complexo (ver Figura 2.1). É válido pontuar que cada bacia de convergência obtida pode ser interpretada como uma rotação da figura gerada pela função $f_{1.1}(x) = x^4 - 1$ (disponível no enunciado do EP):

$z^4 = -1$ (raiz de $f_1(x)$), $w^4 = 1$ (raiz de $f_{1.1}(x)$), $\iff$
$z^4 = -w^4 \iff z^4 = \text{cis}\,(\pi) \cdot w^4 \iff$
$z = \text{cis}\left(\frac{\pi}{4}\right) \cdot w$,

isto é, z é obtido por uma rotação de w em 45° no sentido anti-horário.
Já na função $f_2(x) = x^5 - 1$, obteve-se 5 zonas fractais no (ver Figura 2.2).

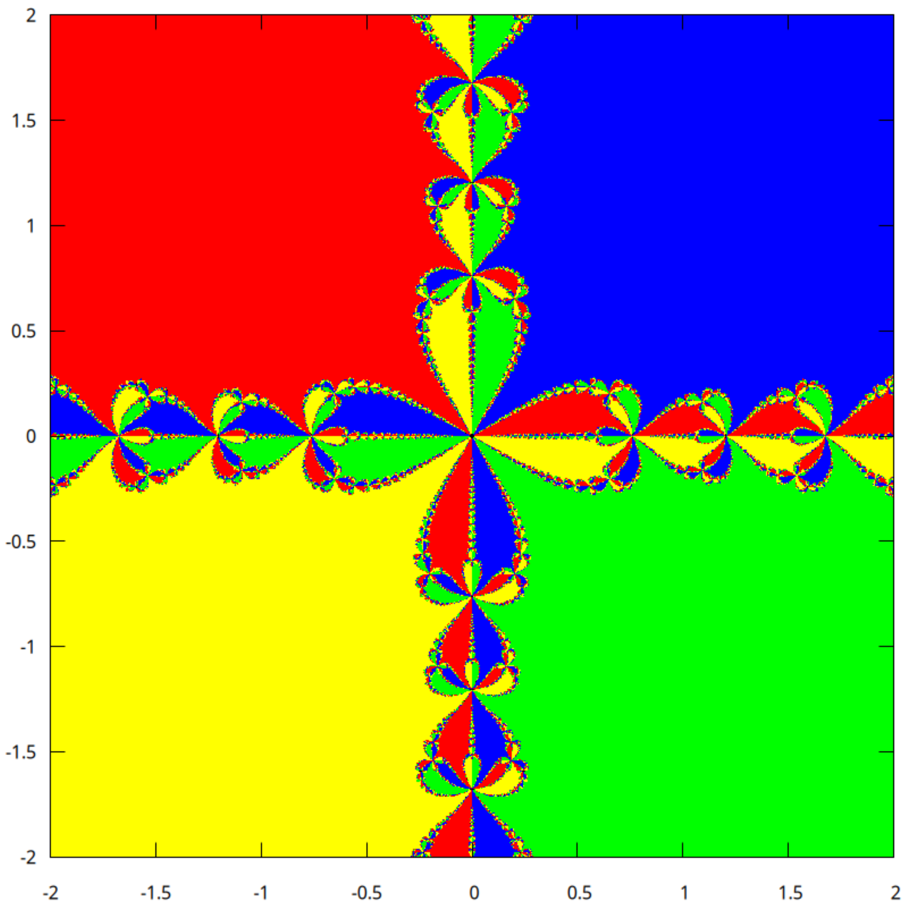


Figura 2.1: Bacias de convergência de $x^4 + 1 = 0$

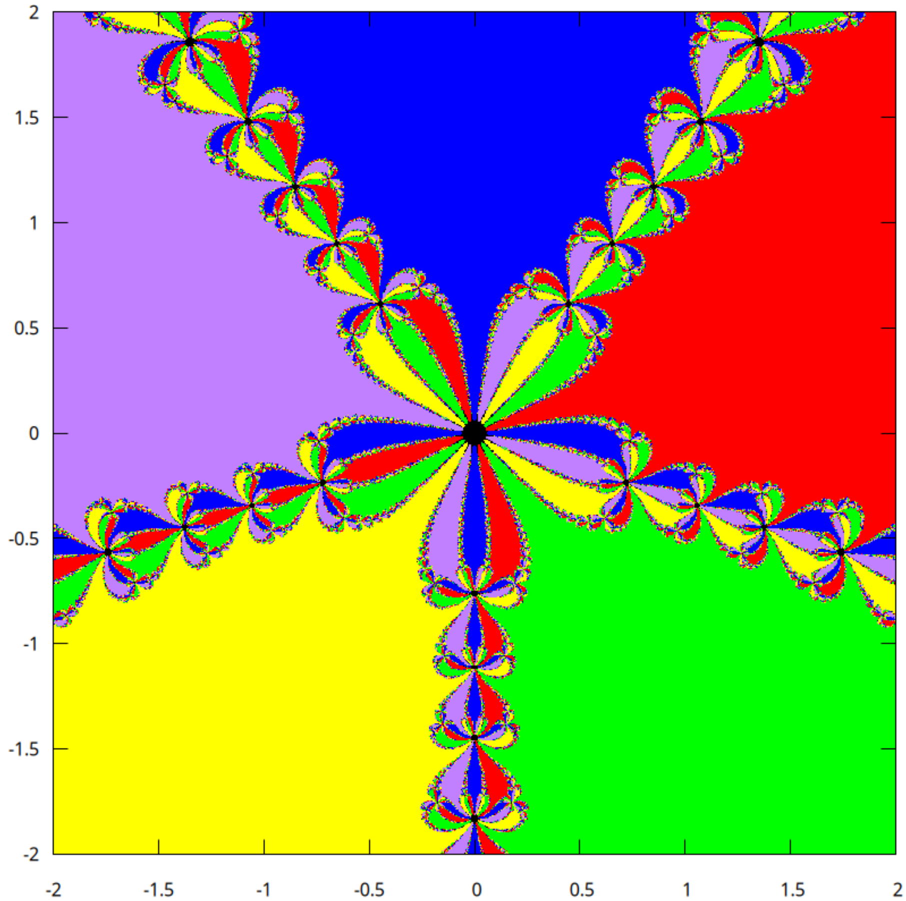


Figura 2.2: Bacias de convergência de $x^5 - 1 = 0$

2.3 Implementação

O código referente à implementação do método de Newton já se encontra comentado e explicado (ver **(EP1_2.c)**). Primeiramente, utilizou-se a biblioteca <*complex.h*> para expandir o método em domínio complexo. A função *newton()*, assim como a *fix()* (método de ponto fixo), possui como critério de parada o erro absoluto *eabs*, além de um controle de máximo de iterações *maxit* em caso de oscilação. Note que a versão atual do programa avalia a função $f_2(x) = x^5 - 1$, representada pelo par (*evalf2*,*evalDf2*). Assim, para analisar a função $f_1(x) = x^4 + 1$, deve-se trocar o par anterior, dentro da função *newton()*, pelo par (*evalf1*,*evalDf1*), e, ainda, alterar o array *root* da função *newton_basins()* de modo a incluir as raízes de $f_1(x)$ (ver **Seção 2.1**), seguida da última alteração que consiste em mudar o 3° *for* (linha 106) para $k < 4$ (raízes z_0, z_1, z_2 e z_3). Nesse caso, a última função citada opera em p pontos (pixels) dentro de um intervalo limitado inferiormente por l e superiormente por u, ou seja, distribui-se simetricamente $\sqrt{p}$ pontos no eixo real e $\sqrt{p}$ pontos no eixo imaginário. Todos os possíveis pares (x, y), cada par representando um número complexo, são usados como ponto de partida em *newton()*, com cada tendência de convergência a uma determinada raiz da função original sendo armazenada em um arquivo *.txt*, o qual será utilizado em *bacia.sh* para plotar as bacias de convergência. Finalmente, para realizar tal plot, então, deve-se primeiro executar *EP1_2.c* para gerar o *.txt* e depois rodar *bacia.sh* para gerar a imagem (lembre-se de usar *chmod +x* previamente). O código também fornece um exemplo de aplicação de $f_2(x)$ na linha 124 para ver na prática sua taxa quadrática (ver Tabela 2).

x_0	**Erros**
1.1	$1.660269107301 \times 10^{-2}$
	$5.335107103572 \times 10^{-4}$
	$5.686605022426 \times 10^{-7}$
	$6.468159341466 \times 10^{-13}$

Tabela 2: Erros por iteração no método de Newton para $f(x) = x^5 - 1$, com $x_0 = 1.1$