pr1_b_math548_final_Lazizbek

May 2, 2024

1 Math 548, Final Exam. Problem 1. LS

Problem 1

(b) Find (3)^(1/2) using the fixed-point iteration method

Solution:

To find $\sqrt{3}$ using the fixed-point iteration method, we need to reformulate the equation $y = \sqrt{3}$ into a fixed-point iteration form $g(y) = y$. We can choose any function $g(y)$ such that its fixed point is $\sqrt{3}$.

I am choosing $g(y) = \frac{1}{2}(y + \frac{3}{y})$, which is derived from rearranging $y = \sqrt{3}$, $y^2 = 3$ as $y = \frac{3}{y}$ and adding both sides y and solving it for y.

Here's the fixed-point iteration method using $g(y)$: 1. Start with an initial guess y_0. 2. Iterate using the formula $y_{n+1} = g(y_n)$. 3. Repeat step 2 until the desired level of accuracy is achieved.

Let's perform the iterations:

$$
\begin{aligned}
&y_0 \text{ (Initial guess)} \\
&y_1 = \frac{1}{2}\left(y_0 + \frac{3}{y_0}\right) \\
&y_2 = \frac{1}{2}\left(y_1 + \frac{3}{y_1}\right) \\
&y_3 = \frac{1}{2}\left(y_2 + \frac{3}{y_2}\right) \\
&\vdots
\end{aligned}
$$

Continuing this process, the sequence converges to $\sqrt{3}$.

Below is the code and results:

[ ]:
```
import pandas as pd

initials = list([2])
steps = list()
approximations = list()
```

```
epsilon = 0.000001

for i in range(len(initials)):
  x0 = initials[i]
  M = 10
  try:
    for k in range(M):
      steps.append(k)
      approximations.append(x0)
      x1 = (x0 + 3/(x0))/2
      if abs(x0-x1) < epsilon:
        print(f"\nWhen x0={initials[i]}, |g'({initials[i]})| < 1, so iteration␣
 ↪converges with tolerance of {epsilon} in {k} steps as follows:")
        break
      x0 = x1


    d = {'step k = ': steps, 'approximation x = ': approximations}
    df = pd.DataFrame(data=d)
    print(df)
    steps = []
    approximations = []

  except:
    print(f"\nWhen x0={initials[i]}, |g'({initials[i]})|>=1, so iteration␣
 ↪diverges in {k} steps as follows:")
    steps.pop()
    approximations.pop()
    d = {'step k = ': steps, 'approximation x = ': approximations}
    df = pd.DataFrame(data=d)
    print(df)
    steps = []
    approximations = []
```

```
When x0=2, |g'(2)| < 1, so iteration converges with tolerance of 1e-06 in 3
steps as follows:
   step k =   approximation x =
0          0           2.000000
1          1           1.750000
2          2           1.732143
3          3           1.732051
```

[ ]:
```
# To download
# !sudo apt-get install texlive-xetex texlive-fonts-recommended␣
 ↪texlive-plain-generic
# !jupyter nbconvert --to pdf /content/pr1_i_math548_midterm_Lazizbek.ipynb
```