E06 — ANN0001 — CCI

Aluno: ENDREW RAFAEL TREPTOW HANG Submeter até: 24/09/2019 23:59hs

Q1 Use o método iterativo de Newton, com estimativa inicial $X^{(1)} = [1.69, 1.17]$, para encontrar a aproximação $X^{(4)}$ da solução do sistema

$$\begin{cases} x_1^2 - 3x_2^2 + 5 = 0 \\ x_1^2 + 2x_2^2 - 5 = 0 \end{cases}$$

a) $[1.00005084, 1.41422694]$

b) $[1.000051, 1.4142271]$

c) $[1.00005146, 1.41422756]$

d) $[1.0000517, 1.4142278]$

e) $[1.00003749, 1.41421359]$

f) $[1.00005327, 1.41422937]$

```
X1
1.69 1.17
X2
1.14085799 1.43970085
X3
1.00869564 1.41443917
X4
1.00003748 1.41421358
```

```python
def F(x):
   return [(x[0]**2 - 3 * x[1]**2 + 5), (x[0]**2 + 2 * x[1]**2 - 5)]


def JF(x):
   yaux1 = [2*x[0], -6*x[1]]
   yaux2 = [2*x[0], 4*x[1]]
   return [yaux1, yaux2]


def inv(x):
   yaux1 = [4*x[1], 6*x[1]]
   yaux2 = [-2*x[0], 2*x[0]]
   return [yaux1, yaux2]


def det(y):
   return y[0][0] * y[1][1] - y[0][1] * y[1][0]
```

```python
x = [1.69, 1.17]

for i in range(3):
    print(f"X{i + 1}")
    print(*[round(i, 8) for i in x])

    r = [0, 0]

    for y in range(2):
        r[y] = inv(x)[y][0] * F(x)[0]
        r[y] += inv(x)[y][1] * F(x)[1]
        r[y] *= (1 / det(JF(x)))

    x[0] -= r[0]
    x[1] -= r[1]

print(f"X4")
print(*[round(i, 8) for i in x])
```