```python
import math

def DD(x, y, yd, n):
    dd = [[0] * n for _ in range(n)]

    for i in range(n):
        dd[i][0] = y[i]
        dd[i][1] = yd[i]

    for j in range(2, n):
        for i in range(n - j):
            dd[i][j] = (dd[i + 1][j - 1] - dd[i][j - 1]) / (x[i + j] - x[i])

    return dd

def Hermite(x, y, yd, n, num):

    dd = DD(x, y, yd, n)
    res = y[0]

    for i in range(1, n):
        prod = dd[0][i]
        for j in range(i):
            prod *= (num - x[j])
        res += prod

    return res


x = [0,0.25]
y = [0,1]
yd = [math.pi , 2*math.pi]
x0 = 0.125
v   = Hermite(x, y, yd, x.__len__(), x0)
print("\nP(", x0, ") = ",round(v,5),"\n")
```

```
  up.edu.mx/Documents/UP/CUARTO SEMES
TRE/CÁLCULO NUMÉRICO/hermite2.py"

P( 0.125 ) =  0.3927

PS C:\Users\luisa\OneDrive - up.edu.
```