Limits

🎯Focus on segmented functions, left and right limits, and singular points

The conceptual framework of this project aligns with core concepts covered in AP calculus BC, especially limits and asymptotic behavior.

Tools 🔧 :

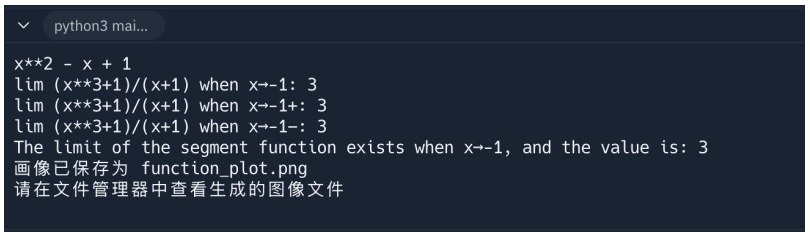
> Matplotlib facilitates the visualization of functions that exhibit non-trivial or irregular behavior.

Limit-example1(limit exists)

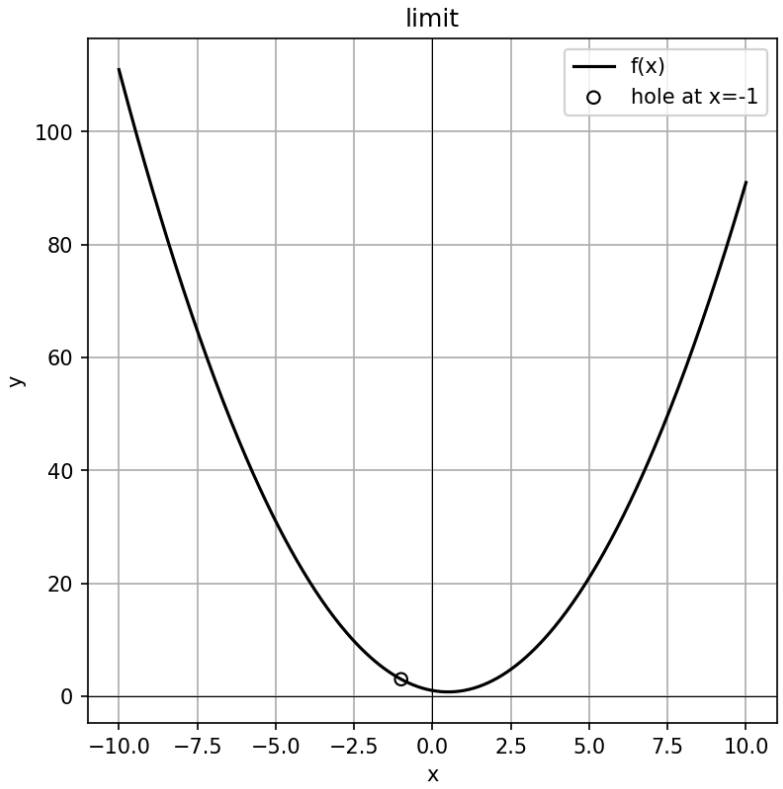
```
from sympy import symbols, limit, oo, simplify, lambdify, sympify
import numpy as np
import matplotlib
matplotlib.use('Agg')
import matplotlib.pyplot as plt
x=symbols('x')
f1=(x**3+1)/(x+1)
func=simplify(f1)
print(func)
lim1=limit(f1,x,-1)
print(f"lim (x**3+1)/(x+1) when x→-1: {lim1}")
lim1_right=limit(f1,x,-1,dir='+')
print(f"lim (x**3+1)/(x+1) when x→-1+: {lim1_right}")
lim1_left=limit(f1,x,-1,dir='-')
print(f"lim (x**3+1)/(x+1) when x→-1−: {lim1_left}")
if lim1_left == lim1_right:
    print("The limit of the segment function exists when x→-1, and the value is:'
else:
    print("The limit of the segmented function does not exist when x→-1")
expr=sympify(f1)
f = lambdify(x, expr, "numpy")
x_vals = np.linspace(-10, 10, 400)
x_vals = x_vals[x_vals != -1]
y_vals = f(x_vals)
plt.figure(figsize=(6,6))
```

```
plt.plot(x_vals, y_vals, label='f(x)', color='black')
y_hole = 3
plt.scatter([-1], [y_hole],
            facecolors='none',
            edgecolors='black',
            label='hole at x=-1')
plt.axhline(0, color='black', linewidth=0.5)
plt.axvline(0, color='black', linewidth=0.5)
plt.title("limit")
plt.xlabel("x")
plt.ylabel("y")
plt.legend()
plt.grid(True)
plt.savefig('function_plot.png', dpi=150, bbox_inches='tight')
plt.close()
print("Saved as function_plot.png")
print("Please view the generated image file in the file manager.")
```

Output from Replit:



Visualization:



Limit-example2(limit doesn’t exist)

```
from sympy import Symbol, limit, oo, sympify, diff, lambdify
import numpy as np
import matplotlib.pyplot as plt
x=Symbol('x')
f3 = 1 / x
lim_pinf = limit(f3, x, 0, dir='+')
print(f"when x→0⁺: {lim_pinf}")
lim_ninf=limit(f3,x,0,dir='-')
print(f"when x→0⁻: {lim_ninf}")

if lim_pinf==lim_ninf:
    print("the limit exists")
else:
    print("the limit doesn't exist")
expr=sympify(f3)
derivative = diff(expr, x)
f = lambdify(x, expr, "numpy")
f_prime = lambdify(x, derivative, "numpy")
```

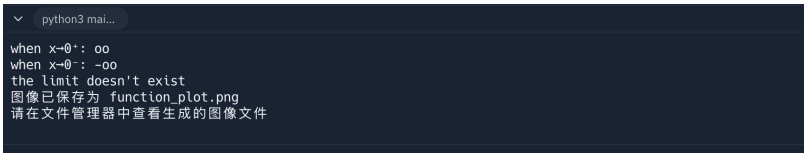
```
x_vals = np.linspace(-10, 10, 400)
y_vals = f(x_vals)
y_prime_vals = f_prime(x_vals)

plt.figure(figsize=(10, 6))
plt.plot(x_vals, y_vals, label="f(x)", color="blue")

plt.axhline(0, color='black', linewidth=0.5)
plt.axvline(0, color='black', linewidth=0.5)

plt.title("Limit")
plt.xlabel("x")
plt.ylabel("y")
plt.legend()
plt.grid(True)
plt.savefig('function_plot.png', dpi=150, bbox_inches='tight')
plt.close()
print("Saved as function_plot.png")
print("Please view the generated image file in the file manager.")
```

Output from Replit:



Visualization :