MATLAB Symbolic Operations: Limits, Derivatives, Integrals and Laplace Transforms

With Commands and Examples

1. Limit

- `limit(expr, var, a)` – Limit at point a

  Example: limit(sin(x)/x, x, 0) → 1

- `limit(expr, var, Inf)` – Limit at infinity

  Example: limit(1/x, x, Inf) → 0

- `limit(expr, var, a, 'left')` – Left-hand limit

  Example: limit(1/(x-1), x, 1, 'left') → -Inf

- `limit(expr, var, a, 'right')` – Right-hand limit

  Example: limit(1/(x-1), x, 1, 'right') → Inf

2. Derivative

- `diff(expr)` – First derivative with respect to default variable
  Example: diff(x^2 + 3*x) → 2*x + 3
- `diff(expr, var)` – First derivative w.r.t specific variable
  Example: diff(x^2*y, x) → 2*x*y
- `diff(expr, var, n)` – n-th order derivative
  Example: diff(sin(x), x, 2) → -sin(x)

3 .Integral

- `int(expr)` – Indefinite integral w.r.t default variable
  Example: int(x^2) → x^3/3
- `int(expr, var)` – Indefinite integral w.r.t specified variable
  Example: int(x*y, x) → (x^2*y)/2
- `int(expr, a, b)` – Definite integral
  Example: int(x^2, 0, 2) → 8/3
- `int(expr, var, a, b)` – Definite integral w.r.t variable
  Example: int(x*y, x, 0, 1) → y/2

4. Laplace Transform

- `laplace(expr)` – Laplace transform with default vars
  Example: laplace(exp(-2*t)) → 1/(s+2)
- `laplace(expr, t, s)` – Laplace transform from t to s
  Example: laplace(sin(t), t, s) → 1/(s^2 + 1)
- `ilaplace(expr)` – Inverse Laplace transform
  Example: ilaplace(1/(s^2 + 1)) → sin(t)
- `ilaplace(expr, s, t)` – Inverse with variables
  Example: ilaplace(1/(s+2), s, t) → exp(-2*t)

Important MATLAB Functions for Calculus and Laplace Transform

Field	MATLAB Function	Description
Limit	**limit(f, x, a)**	**Calculate limit of f(x) as x approaches a**
Right-hand Limit	**limit(f, x, a, 'right')**	**Calculate limit from the right**
Left-hand Limit	**limit(f, x, a, 'left')**	**Calculate limit from the left**
First Derivative	**diff(f, x)**	**Compute first derivative of f(x)**
Second Derivative	**diff(f, x, 2)**	**Compute second derivative of f(x)**
Partial Derivative	**diff(f, var)**	**Compute partial derivative for multivari**
Indefinite Integral	**int(f, x)**	**Compute indefinite integral of f(x)**
Definite Integral	**int(f, x, a, b)**	**Compute definite integral from a to b**
Laplace Transform	**laplace(f, t, s)**	**Compute Laplace Transform of f(t)**
Inverse Laplace	**ilaplace(F, s, t)**	**Compute Inverse Laplace Transform ba**

Quick MATLAB Examples

Calculate Limit:

```
syms x
limit(sin(x)/x, x, 0)
```

First Derivative:

```
syms x
diff(x^3 + 2*x, x)
```

Indefinite Integral:

```
syms x
int(x^2, x)
```

Laplace Transform:

```
syms t s
laplace(exp(-2*t), t, s)
```

Exercises on Limits, Derivatives, ODEs, Integration, and Laplace Transform using MATLAB

Exercise 1: Calculating a Limit

Task:

Use MATLAB to calculate:

$\lim(x \to 0) (\sin(x)/x)$

MATLAB Solution:

```
syms x
limit(sin(x)/x, x, 0)
```

Exercise 2: Derivative of a Polynomial

Task:

Find the first and second derivatives of f(x) = x^3 + 2x^2 + x using MATLAB.

MATLAB Solution:

```
syms x
f = x^3 + 2*x^2 + x;
df = diff(f)
d2f = diff(f, 2)
```

Exercise 5: Indefinite and Definite Integration

Task:

Find the indefinite and definite integral of x^2 from 0 to 2.

MATLAB Solution:

```
syms x
int(x^2)
int(x^2, x, 0, 2)
```

Exercise 6: Laplace Transform

Task:

Compute the Laplace transform of f(t) = exp(2t) using MATLAB.

MATLAB Solution:

```
syms t s
f = exp(2*t);
F = laplace(f, t, s)
```

Exercise 7: Inverse Laplace Transform

Task:

Find the inverse Laplace of 1/(s^2 + 4).

MATLAB Solution:

```
syms s t
f = ilaplace(1/(s^2 + 4), s, t)
```

Practical Exercises – Laplace Transform in MATLAB

Exercise 1: Laplace Transform of a Polynomial

Problem:

Find the Laplace transform of f(t) = t^2 + 3t + 1.

Solution:

```
syms t s
f = t^2 + 3*t + 1;
F = laplace(f)
```

Exercise 2: Laplace Transform of an Exponential Function

Problem:

Find the Laplace transform of f(t) = e^(2t).

Solution:

```
syms t s
f = exp(2*t);
F = laplace(f)
```

Exercise 3: Laplace Transform of a Sine Function

Problem:

Find the Laplace transform of f(t) = sin(3t).

Solution:

```
syms t s
f = sin(3*t);
F = laplace(f)
```

Exercise 4: Laplace Transform of a Piecewise Function

Problem:

Find the Laplace transform of f(t) = heaviside(t - 2).

Solution:

```
syms t s
f = heaviside(t - 2);
F = laplace(f)
```

Exercise 5: Inverse Laplace Transform

Problem:

Find the inverse Laplace transform of F(s) = 1/(s^2 + 4).

Solution:

```
syms s t
F = 1/(s^2 + 4);
f = ilaplace(F)
```

Exercise 6: Laplace of Cosine Function

Problem:

Find the Laplace transform of f(t) = cos(5t).

Solution:

```
syms t s
f = cos(5*t);
F = laplace(f)
```

Exercise 7: Laplace of Derivative

Problem:

Find the Laplace transform of f'(t), where f(t) = sin(t).

Solution:

```
syms t s
f = sin(t);
df = diff(f, t);
F = laplace(df)
```

Exercise 8: Laplace of Second Derivative

Problem:

Find the Laplace transform of d^2f/dt^2 where f(t) = e^(-t).

Solution:

```
syms t s
f = exp(-t);
d2f = diff(f, t, 2);
F = laplace(d2f)
```

Exercise 9: Laplace with Initial Conditions

Problem:

Find the Laplace of f''(t) with f(0)=2 and f'(0)=0, f(t)=t^2.

Solution:

```
syms t s f(t)
d2f = diff(f, t, 2);
F = laplace(d2f, t, s);
F = subs(F, [f(0), diff(f(t), t, 1)(0)], [2, 0])
```

Exercise 10: Inverse Laplace with Shift

Problem:

Find the inverse Laplace transform of F(s) = exp(-2*s)/s.

Solution:

```
syms s t
F = exp(-2*s)/s;
f = ilaplace(F)
```

Simple Solved Examples in MATLAB: if, for, while

Example 1: Using if statement

Task:

Check if a number is positive, negative, or zero.

MATLAB Code:

```
x = input('Enter a number: ');

if x > 0
   disp('The number is positive.')
elseif x < 0
   disp('The number is negative.')
else
   disp('The number is zero.')
end
```

Example 2: Using for loop

Task:

Print numbers from 1 to 5.

MATLAB Code:

```
for i = 1:5
   disp(i)
end
```

Example 3: Using for loop to calculate sum

Task:

Calculate the sum of numbers from 1 to 10.

MATLAB Code:

```
sum = 0;
for i = 1:10
```

```
    sum = sum + i;
end
disp(['Sum = ', num2str(sum)])
```

Example 4: Using while loop

Task:

Print numbers from 1 to 5 using a while loop.

MATLAB Code:

```
i = 1;
while i <= 5
    disp(i)
    i = i + 1;
end
```

Example 5: Using while to check user input

Task:

Keep asking the user to input the number 5 until they do.

MATLAB Code:

```
number = 0;
while number ~= 5
    number = input('Enter the number 5: ');
end
disp('Correct! You entered 5.')
```

Example 6: Using if and for together

Task:

Print even numbers from 1 to 10.

MATLAB Code:

```
for i = 1:10
    if mod(i,2) == 0
        disp(i)
    end
end
```

1. Right-angle Triangle (for)

```
rows = 5;
for i = 1:rows
    for j = 1:i
        fprintf('* ');
    end
    fprintf('\n');
end
```

2. Square (while)

```
n = 4;
i = 1;
while i <= n
    j = 1;
    while j <= n
        fprintf('* ');
        j = j + 1;
    end
    fprintf('\n');
    i = i + 1;
end
```

3. Inverted Triangle (for)

```
rows = 5;
for i = rows:-1:1
    for j = 1:i
        fprintf('* ');
    end
    fprintf('\n');
end
```

4. Right-aligned Triangle (for + if)

```
rows = 5;
for i = 1:rows
    for j = 1:rows
        if j <= rows - i
            fprintf('  ');
        else
            fprintf('* ');
        end
    end
    fprintf('\n');
end
```

5. X Shape (for + if)

```
n = 5;
for i = 1:n
    for j = 1:n
        if i == j || j == (n - i + 1)
```

```
            fprintf('*');
        else
            fprintf(' ');
        end
    end
    fprintf('\n');
end
```

6. Rectangle (for)

```
rows = 3;
cols = 6;
for i = 1:rows
    for j = 1:cols
        fprintf('* ');
    end
    fprintf('\n');
end
```

7. Ladder (while)

```
n = 5;
i = 1;
while i <= n
    j = 1;
    while j <= i
        fprintf('* ');
        j = j + 1;
    end
    fprintf('\n');
    i = i + 1;
end
```

8. Letter L (for + if)

```
rows = 5;
cols = 4;
for i = 1:rows
    for j = 1:cols
        if i == rows || j == 1
            fprintf('* ');
        else
            fprintf('  ');
        end
    end
    fprintf('\n');
end
```

9. Left-leaning Triangle (for + if)

```
rows = 5;
for i = 1:rows
    for j = 1:rows
        if j >= i
```

```
            fprintf('* ');
        else
            fprintf('  ');
        end
    end
    fprintf('\n');
end
```

10. Angle Shape (for + if)

```
rows = 5;
for i = 1:rows
    for j = 1:i
        if j == 1 || i == rows || j == i
            fprintf('* ');
        else
            fprintf('  ');
        end
    end
    fprintf('\n');
end
```

1. Right-angle Triangle (for)

```
rows = 5;
for i = 1:rows
    for j = 1:i
        fprintf('* ');
    end
    fprintf('\n');
end
```

2. Square (while)

```
n = 4;
i = 1;
while i <= n
    j = 1;
    while j <= n
        fprintf('* ');
        j = j + 1;
    end
    fprintf('\n');
    i = i + 1;
end
```

3. Inverted Triangle (for)

```
rows = 5;
for i = rows:-1:1
    for j = 1:i
        fprintf('* ');
    end
    fprintf('\n');
end
```

4. Right-aligned Triangle (for + if)

```
rows = 5;
for i = 1:rows
    for j = 1:rows
        if j <= rows - i
            fprintf('  ');
        else
            fprintf('* ');
        end
    end
    fprintf('\n');
end
```

5. X Shape (for + if)

```
n = 5;
for i = 1:n
    for j = 1:n
        if i == j || j == (n - i + 1)
```

10 MATLAB Questions with Solutions (if, for, while)

Question 1: Check if a number is even or odd.

```
x = input('Enter a number: ');
if mod(x,2) == 0
    disp('Even number')
else
    disp('Odd number')
end
```

Question 2: Display multiples of 3 between 1 and 15.

```
for i = 1:15
    if mod(i,3) == 0
        disp(i)
    end
end
```

Question 3: Sum numbers starting from 1 until sum exceeds 50.

```
sum = 0;
i = 1;
while sum <= 50
    sum = sum + i;
    i = i + 1;
end
disp(['Sum = ', num2str(sum)])
```

Question 4: Find factorial of a number using for loop.

```
n = input('Enter a number: ');
fact = 1;
for i = 1:n
    fact = fact * i;
end
disp(['Factorial = ', num2str(fact)])
```

Question 5: Check if grade is Pass or Fail.

```
grade = input('Enter your grade: ');
if grade >= 50
    disp('Pass')
else
    disp('Fail')
end
```

Question 6: Print first 5 square numbers.

```
for i = 1:5
    disp(i^2)
end
```

Question 7: Keep asking user to enter positive number.

```
num = -1;
while num <= 0
    num = input('Enter a positive number: ');
end
disp('Correct input.')
```

Question 8: Display numbers divisible by 2 and 3 between 1 and 20.

```
for i = 1:20
    if mod(i,2) == 0 && mod(i,3) == 0
        disp(i)
    end
end
```

Question 9: Print numbers from 10 down to 1.

```
i = 10;
while i >= 1
    disp(i)
    i = i - 1;
end
```

Question 10: Calculate product of numbers from 1 to 5.

```
prod = 1;
for i = 1:5
    prod = prod * i;
end
disp(['Product = ', num2str(prod)])
```