

---

```

function [price] = MertonDigitalEurOptPricing ...
    (lambda, sigma, alpha, beta, a, S0, T, eta, L)
% Computes price for digital option in Merton model using Fourier
% pricing formula

gamma = -0.5 * sigma ^ 2 - lambda * (exp(alpha + 0.5 * beta ^ 2) - 1);

% characteristic function of the Fourier transform
FP = @(u) exp(T * ( 1i * gamma * u - 0.5 * sigma^2 * u.^2 + ...
    lambda * (exp(1i * alpha * u - 0.5 * beta^2 * u.^2) - 1)));

% payoff Fourier transform
Fg = @(u) 1i * exp(1i * log(a) * u) ./ u;

% integrand function
int = @(u) real(exp(1i * log(S0) * u) .* FP(u + 1i * eta) .* Fg(-u -
    1i * eta));

% pricing formula
price = exp(-eta * log(S0)) / pi * integral(int,0,L);

end

```

*Not enough input arguments.*

*Error in MertonDigitalEurOptPricing (line 5)*

```
gamma = -0.5 * sigma ^ 2 - lambda * (exp(alpha + 0.5 * beta ^ 2) - 1);
```

*Published with MATLAB® R2020a*