

---

```
function [C] = joshPrincAxe2RotM(a,phi)
arguments
    a (1,3) {mustBeReal, mustBeNumeric}
    phi (1,1) {mustBeReal, mustBeNumeric}
end
if abs(norm(a) - 1) > 1e-14 % checks if a is a unit vector, rounds so that an
    error near e-mach will not cause a failure
    throw(MException("joshPrincAxe2RotM:invalidInput","a is not a unit
        vector"))
end
[n,m] = size(a);
if (n == 3 & m == 1) | (n == 1 & m == 3) % a must be a 1x3 or 3x1
    if m == 3
        a = a'; % if a is a horizontal vector it will be transposed to
            vertical
    end
else % a must have 3 components
    throw(MException("joshPrincAxe2RotM:invalidInput","a is not a 1x3 or
        3x1"))
end

ax = joshCross(a); % a-"cross", returns a symbolic type
ax = double(ax); % ax cast to double from symbolic

x=ax(1); % short hands for readability
y=ax(2);
z=ax(3);
c=cos(phi);
s=sin(phi);
Cs = 1-c; % shortHand usually called C

C = c*eye(3)+Cs*(a*a')-s*ax; % calculates C in terms of ax and phi

end
```

*Published with MATLAB® R2022a*