```
function [fz,y,A,z,flag,glag,gdotlag] =
 joshfLambert(r1,r2,dt,theta,mu,C,S)%,z0)%,pr)
%{
this function will return the lagrange coeffs along with z from the
universal variable, A and yfrom lamberts problem and fz from lamberts
problem such that the zero of fz will give you the solution z to lamberts
problem. r1 r2 should be scalars, dt is transit tim
%}


arguments
    r1 (1,1) double {mustBeReal, mustBePositive}
    r2 (1,1) double {mustBeReal, mustBePositive}
    dt (1,1) double {mustBePositive}
    theta (1,1) double {mustBePositive}
    mu (1,1) double {mustBePositive} = 398600
    C (1,:) double {mustBeReal} = nan
    S (1,:) double {mustBeReal} = nan
    %     z0 (1,1) double {mustBeReal} = nan
end


warning("joshfLambert: This function may be useful but it is not well tested
 and complete argument validation has not been implimented.")


if isnan(C)|isnan(S)
    [C,S] = joshStumpffCoeffs();
end

coefs = length(C);
if length(S)~=coefs
    throw(MException("joshfLambert:invalidInput","S and C should be the same
 length"))
end

A = sin(theta)*sqrt(r1*r2/(1 - cos(theta)));

% disp("josh")
% disp(theta)
% disp(r1)
% disp(r2)
% disp(A)

C = @(z) sum(C.*joshStumpffZ(z,coefs));
S = @(z) sum(S.*joshStumpffZ(z,coefs));

y = @(z)(r1+r2+  A*((z*S(z)-1))/sqrt(C(z))); % y is correct
fz = @(z) S(z)*(y(z)/C(z))^(1.5)+  A*sqrt(y(z))-sqrt(mu)*dt; % f is correct
z = fzero(fz,0);

flag = 1-(y(z)/norm(r1));
```

```
glag = A*sqrt(y(z)/mu);
% fdot = (sqrt(mu_e)/(norm(r1)*norm(r2)))*sqrt(y(z)/C(z))*(z*S(z)-1);
gdotlag = (1-(y(z)/norm(r2)));

% my attempts
%WRONG:
% fp = @(z) (1/(2*sqrt(y(z)*C(z)^5)) )  *
 ( y(z)^2*((2*C(z)*dS(z))-3*dC(z)*S(z)) + dy(z)*(A*C(z)^(5/2) +
 3*C(z)*S(z)*y(z)) );
% dC = @(z) (1/(2*z)) * (C(z)-3*S(z));
% dS = @(z) (1/(2*z)) * (1-z*S(z)-2*C(z));
% dy = @(z)(A/(2*C(z)^1.5)) * ( (dC(z)*(1-z*S(z)) +  2*C(z)*(S(z)+z*dS(z))) );
% fpz0 = @(z)  sqrt(2)/40*y(0)^1.5 + A/8*(sqrt(y(0)) + A*sqrt(1/2/y(0)));
%
% fp = @(z) (y(z)/C(z))^1.5*(1/2/z*(C(z) - 3*S(z)/2/C(z))+ 3*S(z)^2/4/C(z)) +
 A/8*(3*S(z)/C(z)*sqrt(y(z))+ A*sqrt(C(z)/y(z)));
%%%%%%%%%%%


end
```

Published with MATLAB® R2022a