```
% Simple two-spin simulation program.
%
% i.kuprov@soton.ac.uk

function hello_world()

% Define Pauli matrices
sigma_x=[0 1/2; 1/2 0];
sigma_y=[0 -1i/2; 1i/2 0];
sigma_z=[1/2 0; 0 -1/2];
unit=[1 0; 0 1];

% Calculate two-spin operators
Lx=kron(sigma_x,unit); Sx=kron(unit,sigma_x);
Ly=kron(sigma_y,unit); Sy=kron(unit,sigma_y);
Lz=kron(sigma_z,unit); Sz=kron(unit,sigma_z);

% Build the Hamiltonian
omega_L=2*pi*200; % 200Hz offset frequency for spin L
omega_S=2*pi*400; % 500Hz offset frequency for spin S
J=pi*40; % 40Hz scalar coupling
H=omega_L*Lz+omega_S*Sz+J*(Lx*Sx+Ly*Sy+Lz*Sz);

% Initial state
rho=Lz+Sz;

% Detection state
coil=(Lx+Sx)+1i*(Ly+Sy);

% Build propagators
P_pulse=expm(-1i*(Ly+Sy)*(pi/2));
time_step=1/norm(H);
P_evol=expm(-1i*H*time_step);

% Simulation, stage 1: pulse
rho=P_pulse*rho*P_pulse';

% Simulation, stage 2: evolution
nsteps=2048;       % number of steps in the simulation
fid=zeros(2048,1); % preallocate the array
for n=1:nsteps
    fid(n)=trace(coil'*rho);
    rho=P_evol*rho*P_evol';
end

% Apodization
window_function=exp(-5*linspace(0,1,2048))';
fid=fid.*window_function;

% Fourier transform with zerofill
spectrum=fftshift(fft(fid,8196));

% Plotting
plot(real(spectrum));
```

```
end
```