```
function L = game_play(L,f,ax,r)
%--------------------------------------------------------------------------
%initialize keyboard callbacks
set(f,'KeyPressFcn',@moving)
set(f,'KeyReleaseFcn',@stoppedmoving)

%--------------------------------------------------------------------------
%initialize balloon
vx = 5*L; %initial vx
vy = -5*L; %initial vy

x1b = 0; %x option 1
x2b = 500; %x option 2
xstart = [x1b, x2b]; %x option array
xb = xstart(r) ; %inital x
yb = 750; %initial y


%initialize popper
xm = 250; %initial x
ym = 0; %initial y
vxm = 0; %initial v

%initialize poppershot
xp = xm; %initial x
yp = ym; %initial y
vyp = 0; %initial v

%initialize plots
%popper plot
hPlot2 = plot(xm,ym,'bo','MarkerFaceColor','b','MarkerSize',25,'Parent',ax);
xlim([0 500])
ylim([0 750])
set(ax,'XColor','none','YColor','none')
set(ax,'color','k')

hold on %ball plot
hPlot1 = plot(xb,yb,'ro','MarkerFaceColor','r','MarkerSize',60,'Parent',ax);
%pause(3)

hold on %target plot
hPlot4 = plot(xb,yb,'ko','MarkerFaceColor','k','MarkerSize',3,'Parent',ax);

hold on %poppershot plot
hPlot3 = plot(xm,yp,'wo','MarkerFaceColor','w','MarkerSize',10,'Parent',ax);
hold off

%--------------------------------------------------------------------------
%gameplay algorithm
dt = 1; %time variable
quit = false;
while ~quit %balloon not popped
    yb = yb+(vy*dt); %y movement
    xb = xb+(vx*dt); %x movement

    if yb <= 0 %hits bottom of axes
      yb = 0; %bottom of axes
      vy = 5*(0.5*L); %positive velocity
    elseif yb >= 750 %hits top of axes
```

```
      yb = 750; %top of axes
      vy = -5*(0.5*L); %negative velocity
    elseif xb <= 0 %hits left wall
      xb = 0; %left wall
      vx = 5*(0.5*L); %positive velocity
    elseif xb >= 500 %hits right wall
      xb = 500; %right wall
      vx = -5*(0.5*L); %negative velocity
    end

    xm = xm + (vxm*dt); %step monkey location
    if xm <= 0 %hits left wall
      xm = 0; %left wall
    elseif xm >= 500 %hits right wall
      xm = 500; %right wall
    end

    yp = yp + (vyp*dt); %poppershot y position
    if yp == ym
    xp = xm; %poppershot x postion
    end
    if yp > 750 %top of figure
        yp = ym; %poppershot position reset
        vyp = 0; %poppershot velocity reset
    end

    set(hPlot1,'XData',xb,'YData',yb) %update balloon
    set(hPlot2,'XData',xm,'YData',ym) %update popper
    set(hPlot3,'XData',xp,'YData',yp) %update poppershot
    set(hPlot4,'XData',xb,'YData',yb) %update target
    drawnow %update visualization

    if (((xp-xb)<3)&&((xp-xb)>=-3)) && (((yp-yb)< 36)&&((yp-yb)>=-36))
        %popshot hits the ball
        quit = true;
        L = L+1; %next level
        close
        disp('Congrats, you made it to the next level!')
        disp('') %spacing
        ballPopperMenu(L)
    elseif (((xm-xb)<40)&&((xp-xb)>=-40)) && (((ym-yb)< 40)&&((ym-yb)>=-40))
        %ball hits the popper
        quit = true;
        L = 1; %reset to level 1
        close
        disp('You lose! Better luck next time.')
        disp('') %spacing
        ballPopperMenu(L)
    end
end
%-------------------------------------------------------------------------
    function moving(~,bData) %popper movement callback 1
        speed = 10*(L*0.5); %velocity definition
        switch bData.Key %key data
            case 'leftarrow' %left arrow pressed
                vxm = -speed; %negative velocity
            case 'rightarrow' %right arrow pressed
                vxm = speed; %positive velocity
            case 'uparrow' %uparrow pressed
                vyp = 36; %positive velocity
```

```
        end
    end

    function stoppedmoving(~,bData) %popper movement callback 2
        switch bData.Key %key data
            case {'leftarrow','rightarrow'} %keys unpressed
                vxm = 0; %no velocity
        end
    end
end
```

Published with MATLAB® R2019b