Problem1
2.

```
function [payoff] = f(S)
    if S >= 20
        payoff =20-0.2;
    else
        payoff = max(0,S-0.2)-max(0,S-20);
    end
end

function [price] = Trinomial(S0,T,n,r)
deltat = T/n;
u = 2;
d = 0.5;

low = r*deltat/(u-1);
high = (u-1+u*r*deltat)/(u^2-1);

pos = 1;

for qu = low:0.0001:high
    qd = 2*(qu-r*deltat);
    qm = 1-qu-qd;
    sigma = qu*(u-(1+r*deltat))^2+(1-qu-qd)*(r*deltat)^2+qd*(d-(1+r*deltat))^2;
    if  (sigma >= 0.5^2) & (sigma <= 0.8^2)
        fprintf('ok')
    else
        continue
    end

    for j = 1:n
        for i = 1:2*j+1
            s(j,i)=S0*u^(max(j+1-i,0))*d^(max(i-1-j,0));
        end
    end

    p = zeros(n,2*n+1);
    for w = 1:2*n+1
        p(n,w) = f(s(n,w));
    end

    for g = n-1:-1:1
        for h = 1:2*g+1
            p(g,h) = (qu*p(g+1,h)+qm*p(g+1,h+1) + qd*p(g+1,h+2))/(1+r*deltat);
```

```
        end
    end
    price(pos) = (qu*p(1,1)+qm*p(1,2)+qd*p(1,3))/(1+r*deltat);
    allq(pos) = qu;
    pos = pos+1;
    fprintf('%f',i)

end
end
```

Problem 2

5.

```
function [payoff] = f2(S)
    if (S >0) & (S <= 1)
        payoff = max(0,S - 0.5);
    else if (S>1) &(S<=1.5)
        payoff = 0.5;
    else
        payoff = max(0,2-S);
    end
end

function [price] = BinomialAmer(S0,T,n,r)
deltat = T/n;
u = 1.1;
d = 0.9;

qu = (1+r*deltat-d)/(u-d);
qd = 1-qu;

for j = 1:n
    for i = 1:j+1
        s(j,i)=S0*u^(j+1-i)*d^(i-1);
        option(j,i)= f2(s(j,i));
    end
end

p = zeros(n,n+1);
for w = 1:n+1
    p(n,w) = option(n,w);
end
for g = n-1:-1:1
    for h = 1:g+1
        p(g,h) = max(option(g,h),(qu*p(g+1,h) + qd*p(g+1,h+1))/(1+r*deltat));
```

```
    end
end
price = max(f2(S0),(qu*p(1,1)+qd*p(1,2))/(1+r*deltat));

end
```