```
function [mc_price] = MCpriceBarrierUODM(r,sigma,Nt,Ns,T,s,K,b)
% Compute the Monte Carlo price

delta_t = T/Nt;          % time interval

S = s*ones(Ns,1);      % initial price

% simulated prices between 0 and T/2
for i = 1:Nt/2
    Z = randn(Ns,1);  % random value following a normal distribution
    S = S.*(1 + r*delta_t + sigma*sqrt(delta_t)*Z);
end


% monitoring T/2:
chech_half_T = S < b;
% chech_helf_T is a boolean vector. If we multiply the result by it,↙
we
% will edit the values which do not pass the check (and set them to↙
0),
% otherwise they will remain unchanged

% simulated prices between T/2 and T
for i = (Nt/2+1):Nt
    Z = randn(Ns,1);  % random value following a normal distribution
    S = S.*(1 + r*delta_t + sigma*sqrt(delta_t)*Z);
end


% MC price
chech_T = S < b;
psi = max(S-K,0).*chech_half_T.*chech_T;
% the first part of the previous equation is the definition of psi,↙
then
% we multiply by the check at T/2 and then the check at T
mc_price = (exp(-r*T))/(Ns)*sum(psi);

end
```