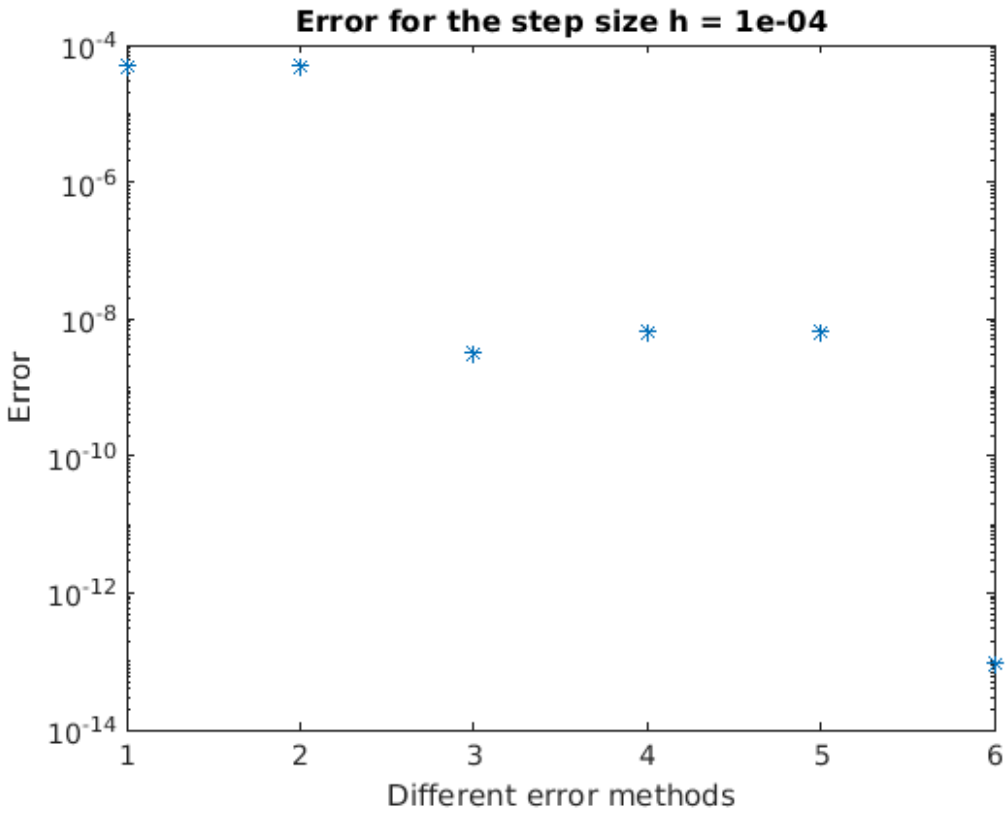
```
% Q2

% Matlab script to calculate
% first order numerical derivative of 2-x+ln(x) at x=1
a = 1;
trueVal = 0;
h = 1e-4;
% Forward diff.
fwdDiff = (f(a + h) - f(a))/h;
errFwd = abs(trueVal - fwdDiff);
disp(['Error in forward diff.: ', num2str(errFwd)]);
% Backward diff.
bckDiff = (f(a) - f(a - h))/h;
errBck = abs(trueVal - bckDiff);
disp(['Error in backward diff.: ', num2str(errBck)]);
% Central diff.
cntrDiff = (f(a + h) - f(a - h))/(2*h);
errCntr = abs(trueVal - cntrDiff);
disp(['Error in central diff.: ', num2str(errCntr)]);
% Improved Forward diff.
iFwdDiff = (-f(a + 2*h) + 4*f(a + h) -3*f(a))/(2*h);
errIFwd = abs(trueVal - iFwdDiff);
disp(['Error in improved forward diff.: ', num2str(errIFwd)]);
% Improved Backward diff.
iBckDiff = (3*f(a) - 4*f(a - h) + f(a - 2*h))/(2*h);
errIBck = abs(trueVal - iBckDiff);
disp(['Error in improved backward diff.: ', num2str(errIBck)]);
% Improved Central diff.
iCntrDiff = (8*f(a + h) - 8*f(a - h) - f(a + 2*h) + f(a - 2*h))/
(12*h);
errICntr = abs(trueVal - iCntrDiff);
disp(['Error in improved central diff.: ', num2str(errICntr)]);
% Plots
semilogy([errFwd, errBck, errCntr, errIFwd, errIBck, errICntr], '*')
title('Error for the step size h = 1e-04')
ylabel('Error')
xlabel('Different error methods')
% Function
function fx = f(x)
  fx = 2 - x + log(x);
end
```

```
Error in forward diff.: 4.9997e-05
Error in backward diff.: 5.0003e-05
Error in central diff.: 3.3334e-09
Error in improved forward diff.: 6.6658e-09
Error in improved backward diff.: 6.668e-09
Error in improved central diff.: 9.2519e-14
```



Published with MATLAB® R2020b