Daniel Hondal

HW 8B: Problem 3

```
fun = @(t) 1./(1+sinh(4*t) + 3*(log(t)).^2)
```

```
fun = function_handle with value:
    @(t)1./(1+sinh(4*t)+3*(log(t)).^2)
```

Part A: Quadrature v. Monte Carlo Integration

```
tic
intval = integral(@(t) fun(t),0,3)
```

```
intval = 0.1129
```

```
t1 = toc
```

```
t1 = 0.1408
```

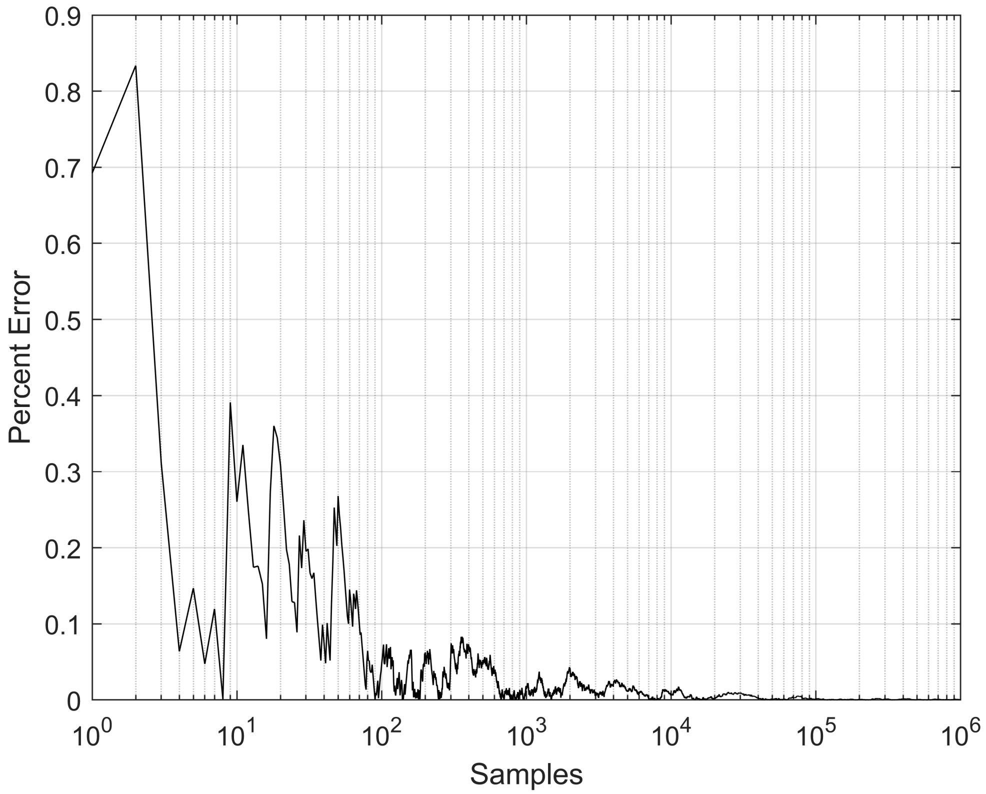
```

tic
xt = 0 + 3*rand(1e6,1);
F = fun(xt);
If = 3*mean(F);
t2 = toc
```

```
t2 = 0.3900
```

```

Fs = cumsum(F);
Ie = 3*Fs./(1:length(Fs))';
semilogx(abs(Ie-intval)/intval,'k-');
grid on;
ylabel('Percent Error')
xlabel('Samples')
```



Quadrature value = 0.1129

MCI value = 0.1129

Part B: Timing

```
formatSpec = 'Quadrature value is %f seconds.';
sprintf(formatSpec,t1)
```

```
ans = 
'Quadrature value is 0.140804 seconds.'
```

```
formatSpec = 'MCI value is %f seconds.';
sprintf(formatSpec,t2)
```

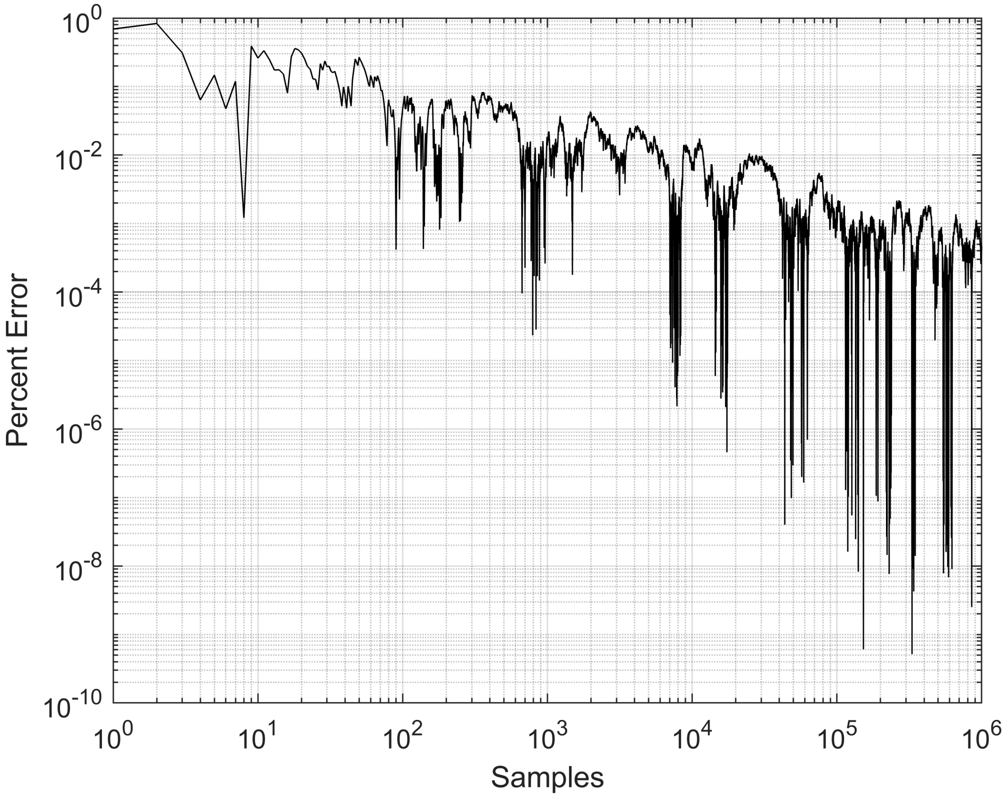
```
ans = 
'MCI value is 0.389983 seconds.'
```

Part C

i. The plot shows the accuracy converging arround 10^3 samples.

ii. As the the order of magnitude of samples increases the accuracy increases exponentially as well (or decreases in error as seen in the log-log plot).

```
loglog(abs(Ie-intval)/intval,'k-');
grid on;
ylabel('Percent Error')
xlabel('Samples')
```


Percent Error
10^{0}
10^{-2}
10^{-4}
10^{-6}
10^{-8}
10^{-10}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
Samples