(b)

In [48]:
```
x = 0.5
n = 3000
u_grid = collect(range(-1, length = n+2, stop = 1)[2 : n+1])
unnormalised_posterior = Array{Float64,1}(undef , n)
delta = u_grid[3]-u_grid[2]
```

Out[48]:
```
0.0006664445184938472
```

In [49]:
```
for i in 1:n
    a =  (u_grid[i]-1)/(u_grid[i]+1)
    b = (1 - u_grid[i])/(u_grid[i]+1)
    unnormalised_posterior[i] = exp(- (x-a)^2/2) * 1/(1+ u_grid[i]^2)*delta+exp(- (x-b)^2/2) * 1/(1+ u_grid[i]^2)*delta
end
```

In [50]:
```
A = sum(unnormalised_posterior)
```

Out[50]:
```
1.549230170781684
```

In [89]:
```
n = [50 250 750 1500 3000]
for o in n
    expectation = Array{Float64,1}(undef , o)
    u_grid = collect(range(-1, length = o+2, stop = 1)[2 : o+1])
    delta = u_grid[3]-u_grid[2]
    for i in 1:o
        a =  (u_grid[i]-1)/(u_grid[i]+1)
        b = (1 - u_grid[i])/(u_grid[i]+1)
        expectation[i] = a*(exp(- (x-a)^2/2) * 1/(1+ u_grid[i]^2)*delta)+b*(exp(- ( x-b)^2/2) * 1/(1+ u_grid[i]^2)*delta)
    end
    print(sum(expectation)/A)
    print('\n')
end
```

```
0.26622665611295937
0.2662266533119162
0.26622665330718864
0.2662266533071297
0.26622665330713763
```

(d)

```
import Pkg  ✓

Pkg.add("StatsBase")  ✓

A = 1.549230170781684  1.55…
x = 0.5  0.500
n = [50 250 750 1500 3000]  > 1×5 Array{Int64,2}:
for o in n
    expectation = Array{Float64,1}(undef , o)
    u_grid = collect(range(-1, length = o+2, stop = 1)[2 : o+1])
    delta = u_grid[3]-u_grid[2]
    for i in 1:o
        a =  (u_grid[i]-1)/(u_grid[i]+1)
        b = (1 - u_grid[i])/(u_grid[i]+1)
        expectation[i] = a*(exp(- (x-a)^2/2) * 1/(1+ u_grid[i]^2
    end
    m = 100
    sample_index = sample(1:o, m, replace = true)
    samples = expectation[sample_index];
    sample_expectation = sum(samples)*n/(m*A)
```

(e)

In [ ]:
```
m = 1000
```

In [66]:
```
x = 0.5; a = -5; b = 5;
```

In [86]:
```
k = [50 250 750 1500 3000]
for n in k

    if n <= 1000
        y_grid = collect(range(a, length=n, stop=b))
       newa = a;
    elseif n > 1000 && n <= 2000
        nm = 1000;
        na=round(Int, (n-nm)/2);
        l = (b - a)/(nm-1);
        newa = a - l*na;
        y_grid = collect( range(newa, step=l , length=n));
    else n > 2000
        nm=round(Int, n/2);
        na=round(Int, (n-nm)/2);
        l = (b - a)/(nm-1);
        newa = a - l*na;
        y_grid = collect(range(newa, step=l , length=n));
    end
    delta = y_grid[3]-y_grid[2]
    unnormalised_posterior = Array{Float64,1}(undef , n)
    for i in 1:n
        unnormalised_posterior[i] =  y_grid[i]*exp(-(x-y_grid[i])^2/2)*(1/(1+y_grid[i]^2))*delta
    end
    print(sum(unnormalised_posterior)/A)
    print('\n')
end
```

```
0.26622606409792576
0.26622573528359345
0.2662256718272629
0.26622665330688033
0.26622665330713263
```