Solution (other solutions are possible and perhaps even better!)

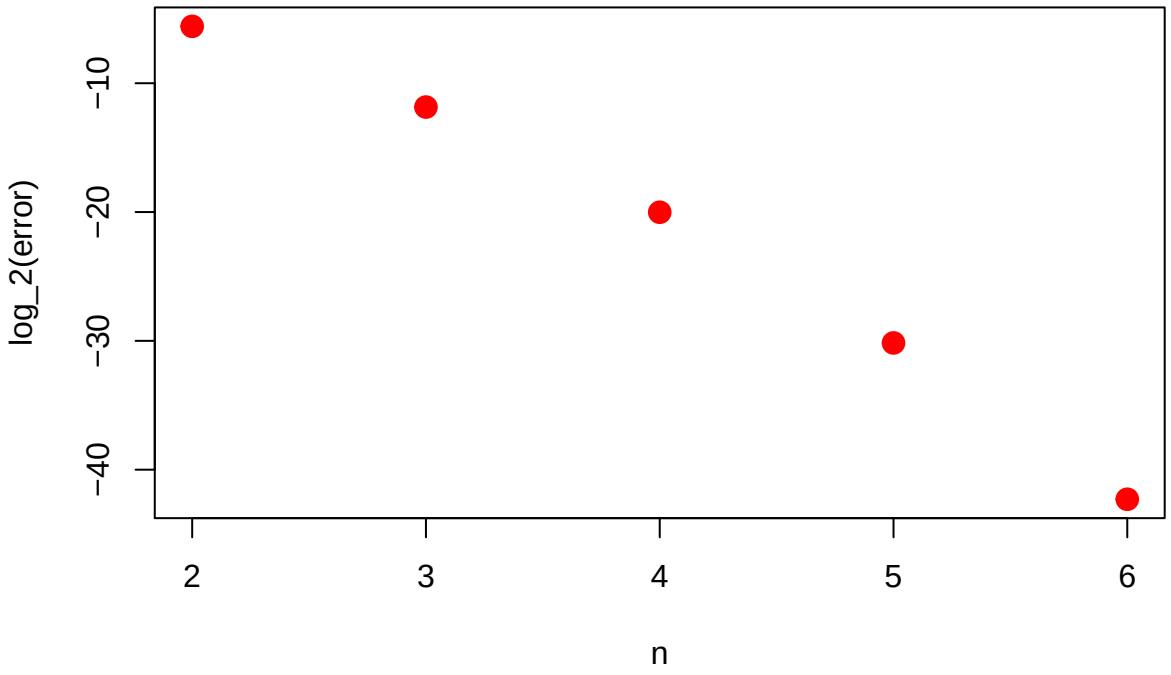
```
rm(list = ls()) # clear memory
cat("\f")       # clear screen

f <- function(x){ exp(-x/2)/2 }

NumInt <- function(f,a,b,n){
  R <- matrix(0,n,n)
  R[1,1] <- (b-a)*(f(a)+f(b))/2
  for(j in 2:n){
    h_j <- (b-a)/(2^(j-1))
    subtotal  <-  0
    for(i in 1:2^(j-2)){
      subtotal  <-  subtotal + f(a+(2*i-1)*h_j)
    }
    R[j,1] <- R[j-1,1]/2+h_j*subtotal;
    for(k in 2:j){
      R[j,k] <- (4^(k-1)*R[j,k-1]-R[j-1,k-1])/(4^(k-1)-1)
    }
  }
  return(R[n,n])
}


approx <- rep(NA,5)
for(i in 1:5){
  n <- i+1
  approx[i] <-  NumInt(f,0,4*log(2)+2*log(5),n)
}
true <- 95/100
error <- abs(true - approx)
for(i in 1:length(error)){
  cat(sprintf("n=%d: True area=%.10f  Approx=%.10f   Error=%.13f\n",i+1,true,approx[i],e
}

plot((2:6),log(error,2),main="Absolute error",xlab="n", ylab="log_2(error)", col="red",
```




```
NumInt2 <- function(b,a,f,n){ NumInt(f,a,b,n)-0.99 }

res <- uniroot(NumInt2,interval=c(8,10),0,f,6)
cat(sprintf("For n=%d and b=%.4f: I(0,b)=0.99.\n",6,res$root))
```

```
##
n=2: True area=0.9500000000  Approx=0.9708305483   Error=0.0208305482757
## n=3: True area=0.9500000000  Approx=0.9502707543   Error=0.0002707543032
## n=4: True area=0.9500000000  Approx=0.9500009430   Error=0.0000009430111
## n=5: True area=0.9500000000  Approx=0.9500000008   Error=0.0000000008335
## n=6: True area=0.9500000000  Approx=0.9500000000   Error=0.0000000000002
## For n=6 and b=9.2103: I(0,b)=0.99.
```