Question\_2.R

Rukmin

Sat Feb 18 22:24:00 2017

# Question 2  
  
SimpsonRule <- function(a,b,m, f){  
 m <- m-1  
 h <- (b-a)/m   
 x <- seq(from = a, to = b, by = h/2)  
 y <- f(x)  
 ix1 <- seq(from =3, by =2, to = 2\*m-1)  
 ix2 <- seq(from =2, by =2, to= 2\*m) - 1  
 return(h/6 \* (y[1] + 2\*sum(y[ix1]) + 4\*sum(y[(ix2)]) + y[2\*m+1]))  
  
}  
  
TrapezoidalRule <- function(a, b, m, f){  
 h <-(b-a)/(m-1)  
 x <- seq(from = a, to = b, length = m)  
 y <- f(x)  
 h \* (0.5 \* y[1] + sum(y[2:(m-1)]) +y[m])  
}  
  
func <- function(x){  
 if (x == 0) {  
 y <- 1  
   
 } else {  
 y <- sin(x) / x  
 }  
 return(y)  
}  
SimpsonRule(-1000000, 1000000, 1000000, func)

## [1] 3.141591

TrapezoidalRule(-1000000, 1000000, 1000000, func)

## [1] 3.141591

SimpsonError <- function (){  
 return(abs(pi – SimpsonRule(-1000000, 1000000, 1000000, func)))  
}

TrapError <- function() {  
 return(abs(pi - TrapezoidalRule(-1000000, 1000000, 1000000, func)))

}  
  
SimpsonError()

## Warning in if (x == 0) {: the condition has length > 1 and only the first  
## element will be used

## [1] 0.000802008

TrapError()

## Warning in if (x == 0) {: the condition has length > 1 and only the first  
## element will be used

## [1] 0.0004516641

#tolerance ----  
newSimpsonRule <- function(a,b,tol,f){  
 m =1000000  
 for(i in 1:m) {  
 temp <- SimpsonRule(a,b,m,f)  
 temp2 <- SimpsonRule(a,b,m+1,f)  
 if (abs(temp-temp2) < tol){  
 return(SimpsonRule(a,b,m,f))  
 }  
 }  
   
}  
newTrapRule <- function(a,b,tol,f){  
 m =1000000  
 for(i in 1:m) {  
 temp <- TrapezoidalRule(a,b,m,f)  
 temp2 <- TrapezoidalRule(a,b,m+1,f)  
 if (abs(temp-temp2) < tol){  
 return(TrapezoidalRule(a,b,m,f))  
 }  
 }  
}  
  
   
  
func2 <- function(x) {  
 return(1 + exp(-x) \* sin(8 \* x^(2/3)))  
}  
  
newSimpsonRule(0,2,1e-4,func2)

## [1] 2.01628

newTrapRule(0,2,1e-4,func2)

## [1] 2.016281