Class Prep 6 | 2.4.1 - 3.1.1

Riley Primeau

2022-10-03

## Section 2.4.1 Simple Division Algorithms

naivediv = function(m, n){  
 quot = 0  
 r = m   
   
 if(n==0)  
 stop("Attempted division by 0")  
   
 while( r>= n) {  
 quot = quot + 1  
 r = r - n  
 }  
 return(list(quotient = quot, remander = r))  
}

naivediv(314, 7)

## $quotient  
## [1] 44  
##   
## $remander  
## [1] 6

floor(314/7)

## [1] 44

314%%7

## [1] 6

longdiv = function(m, n){  
 quot = 0  
 r = 0  
   
 if(n == 0)  
 stop("Attempted division by 0")  
   
 for(i in 31:0){  
 r = bitwShiftL(r,1)  
 r = r + bitwAnd(bitwShiftR(m,i), 1)  
 if(r >= n) {  
 r = r - n  
 quot = quot + bitwShiftL(1, i)  
 }  
 }  
 return(list(quotient = quot, remainder = r))  
}  
  
  
longdiv(314, 7)

## $quotient  
## [1] 44  
##   
## $remainder  
## [1] 6

## Section 3.1.1 Vectir and Matrix Operations

u = c(1, 2, 3); v = c(8, 4, 2); x = 7  
u + x

## [1] 8 9 10

u + v

## [1] 9 6 5

u + c(1,9)

## Warning in u + c(1, 9): longer object length is not a multiple of shorter object  
## length

## [1] 2 11 4

A = matrix(1:9, 3)  
A + 1

## [,1] [,2] [,3]  
## [1,] 2 5 8  
## [2,] 3 6 9  
## [3,] 4 7 10

A + c(1, 2, 3)

## [,1] [,2] [,3]  
## [1,] 2 5 8  
## [2,] 4 7 10  
## [3,] 6 9 12

A + 1

## [,1] [,2] [,3]  
## [1,] 2 5 8  
## [2,] 3 6 9  
## [3,] 4 7 10

A + c(1,2) - A

## Warning in A + c(1, 2): longer object length is not a multiple of shorter object  
## length

## [,1] [,2] [,3]  
## [1,] 1 2 1  
## [2,] 2 1 2  
## [3,] 1 2 1

A + c(1, 2, 3) - A

## [,1] [,2] [,3]  
## [1,] 1 1 1  
## [2,] 2 2 2  
## [3,] 3 3 3

B = matrix(1:6, 3)  
status = try(A+B)

## Error in A + B : non-conformable arrays

print(status[1])

## [1] "Error in A + B : non-conformable arrays\n"

A %\*% B

## [,1] [,2]  
## [1,] 30 66  
## [2,] 36 81  
## [3,] 42 96

u %\*% v

## [,1]  
## [1,] 22

diag(A)

## [1] 1 5 9

diag(B)

## [1] 1 5

diag(u)

## [,1] [,2] [,3]  
## [1,] 1 0 0  
## [2,] 0 2 0  
## [3,] 0 0 3

diag(1,4)

## [,1] [,2] [,3] [,4]  
## [1,] 1 0 0 0  
## [2,] 0 1 0 0  
## [3,] 0 0 1 0  
## [4,] 0 0 0 1

nrow(B)

## [1] 3

ncol(B)

## [1] 2

nrow(u)

## NULL

ncol(u)

## NULL

length(u)

## [1] 3

length(B)

## [1] 6

dim(B)

## [1] 3 2