**-------------------- CntMtx --------------------**

**z809\_1** -------------------------------------------------------------------------------

rawFV :

- nextRebNum : fCutU.hasPtn(c(15,NA,23),aZoid)

auxZW :

- nextRebNum : *36* 40 *35* 35 <34>

- nextColVal\_1 : *32* 31 20 *33* 31 28 <34>

auxQuo :

- nextColVal\_3 : 1 3 1 0 (**unique**: 1 2 1 2 0)

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z814\_1** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

- nextRebNum : all(quoSize[1:3+1]==c(0,2,1))

cStep.w1 :

- nextQuo10 : fCutU.hasPtn(c( 4, 3 ),aCStep)

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

- nextZW : aFStep[3]==sum(aFStep[c(1,2)])

**z816\_0** -------------------------------------------------------------------------------

rawFV :

- nextColVal\_3 : fCutU.hasPtn(c(18,19),aZoid) (\*2)

auxZW :

-

auxQuo :

- nextZW : all(quoSize[1:3+2]==c(1,2,0))

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z819\_\_** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z820\_2** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

- basic : all(aCStep[3:4]==c( 8, 5)) # 3,1 아래 패턴 반복

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

- nextColVal\_2 : aFStep[3]==sum(aFStep[c(1,2)])

**z821\_\_** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z822\_\_** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z823\_1** -------------------------------------------------------------------------------

rawFV :

- nextZW : fCutU.hasPtn(c( 12,NA,24 ),aZoid) # 12 18 24 26 39 40

auxZW :

-

auxQuo :

-

cStep.w1 :

- nextBin : fCutU.hasPtn(c( 6, 6),aCStep) # 6 6 2 13 1

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

- nextColVal\_2 : aFStep[2]==sum(aFStep[c(3,6)])

**z824\_\_** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z825\_0** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

- nextCStepBin : all(quoSize[1:3+0]==c(1,1,1))

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

- nextQuo10 : aFStep[2]==sum(aFStep[c(1,6)])

- nextRebNum : aFStep[1]==sum(aFStep[c(5,6)])

**z826\_1** -------------------------------------------------------------------------------

rawFV :

- nextQuo10 : fCutU.hasPtn(c(13,NA,NA,25 ),aZoid) # 13 16 24 25 33 36

auxZW :

-

auxQuo :

- nextColVal\_2 : all(quoSize[1:3+2]==c(2,2,0)) ( **unique** : 1 1 2 2 )

all(quoSize[1:3+0]==c(1,1,2))

cStep.w1 :

- nextFStepBin : fCutU.hasPtn(c( 3, 8),aCStep) # 3 8 1 8 3

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

- nextCStepBin : aFStep[4]==sum(aFStep[c(3,6)])

- nextColVal\_4 : aFStep[2]==sum(aFStep[c(3,5,6)])

- nextColVal\_6 : aFStep[2]==sum(aFStep[c(3,5,6)])

**z827\_\_** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z828\_\_** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z829\_1** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z830\_1** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

- nextColVal\_5 : 23 25 26 33 *31 32* and 33

auxQuo :

-

cStep.w1 :

- nextColVal\_4 : fCutU.hasPtn(c( 1,10),aCStep) #

cStep.w2 :

- nextQuo10 : aCStep[3]==sum(aCStep[c(1,5)])

fStep.w1 :

-

fStep.w2 :

- nextQuo10 : aFStep[4]==sum(aFStep[c(2,6)])

- nextColVal\_4 : aFStep[3]==sum(aFStep[c(1,2)])

**z831\_\_** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z832\_1** -------------------------------------------------------------------------------

rawFV :

- nextQuo10 : fCutU.hasPtn(c(13,14),aZoid) \* 2 # 13 14 19

fCutU.hasPtn(c(14,19),aZoid)

auxZW :

- nextFStepBin : 20 21 *28* 25 *29* 42 and 30

auxQuo :

-

cStep.w1 :

- nextQuo10 : fCutU.hasPtn(c(1,5),aCStep)

cStep.w2 :

- nextFStepBin : sum(aCStep[c(1,3)])==sum(aCStep[c(2,5)])

fStep.w1 :

-

fStep.w2 :

- nextFStepBin : sum(aFStep[c(1,6)])==sum(aFStep[c(4,5)])

**z833\_\_** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z834\_\_** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z835\_\_** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z836\_1** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

- basic : fCutU.hasPtn(c(8,2),aCStep)

- nextZW : fCutU.hasPtn(c(8,2),aCStep)

- nextColVal3 : fCutU.hasPtn(c(8,2),aCStep)

-

cStep.w2 :

- nextColVal\_1 : aCStep[4]==sum(aCStep[c(1,2,5)])

fStep.w1 :

-

fStep.w2 :

- basic : sum(aFStep[c(1,5)])==sum(aFStep[c(2,6,3)])

- nextColVal\_3 : aFStep[3]==sum(aFStep[c(1,5)])

- nextColVal\_6 : sum(aFStep[c(3,6)])==sum(aFStep[c(4,5)])

**z837\_\_** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z838\_0** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z839\_\_** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z840\_\_** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z840\_\_** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z841\_\_** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z842\_\_** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z843\_\_** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-

**z844\_\_** -------------------------------------------------------------------------------

rawFV :

-

auxZW :

-

auxQuo :

-

cStep.w1 :

-

cStep.w2 :

-

fStep.w1 :

-

fStep.w2 :

-