|  |  |
| --- | --- |
| **Mickl** | **ICMS** |
| MatrixHeight(A) | ncol(A)  (columns and rows are inverted) |
| // | # |
| /\* \*/ | # |
| CreateMatrix(A,ncol,nrow,DOUBLEPRECISION,ERASE); | A=matrix(0,ncol=ncol,nrow=nrow)  TODO: NOERASE |
| \_A | A (underscore not allowed)  Use <<- to assign to global variable previously defined |
| Procedural functions | Either:   * Use <<- to assign to global variable * Copy-paste * Transform to proper function |
| for (I = 1800; I <= y; i++) { | for(I in 1800:y){ |
| mod(a,b) | %% (double-precision modulus)  round (rounding) |
| return 0; | return(a);  No need to return formally. |
| function num\_days(d,m,y){ | num\_days <- function(d,m,y){ |
| A[col,row] | A[row,col]  It may be easier to transpose the data using t(M) and keep the code the same |
| integer | Conversion to integer not usually needed, but uses integer  Rounding uses round |
| Power(a,b) | a^b |
| Round | round |

Can be kept but aren’t necessary

* Final commas ;
* Decimal points on integers to make them double precision, e.g. 4.