**Main Function()**

*Lines 84-104: Check arguments inputted from command line. If there is no argument, invoke an error. If there is one argument, set is as input file (usually mps format). If there are 3 arguments, the first should be -s, the second will be the name of the specs file, and the last will be the name of the input file (usuallly MPS)*

*LIne 112: Set default input values in variable inputs by calling the NewParameters() function in the parameters.c file.*

*Lines 115-118: alter inputs by reading for specs file. Check if errors where generated when reading specs file. Uses ParseSpecsFile() and CheckParameters() from parameters.c*

*Lines 125-131:Read in file data and put Into file pointer fp. Uses OpenInputFile from io.c.*

*Lines 135-141: Read in MPS file using file pointer and Specs inputs and check for error. Uses ReadMPS in readmps.c*

*Lines 151-157: Convert to LP type and print LP formultion. Uses Convert\_MPS\_LP() in lpmps.c*

*Line 159: Get start time for preprocessing. Uses GetTime from timers.c*

*Lines 161- 176: apply preprocessing to get reduced LP and check for error. Uses Preprocess() from presolve.c and ScaleLP from scale.c*

*Lines 180-182: calculate preprocessing time*

*Lines 187- 190: Setup solution data structure*

*Line 194: get time for start of solving*

*Line 196-204: compute solution and time duration. Uses PCx() from PCx.c*

*Lines 210-217: recover free variable values. Uses UnSplitFreeVars() from split.c*

*Line 220: Check infeasibilities and print. Uses ComputeAndPrintInfeasibilities() from io.c*

*Lines 223-227: Express the solution in MPS formulation: uses MPSsolution() from lpmps.c*

*Line 230: prints the solution. Uses PrintSolution() from io.c*

*Lines 232-235: free data*

*Line 238: Exit*