GR / GRX © Copyright Sharrad Software 1997-2011

Contact support@sharradsoftware.co.uk or visit www.sharradsoftware.co.uk

```
Version History:
  1.06  10-08-1997
  1.07  18-08-1997  Change to using one file handle for open and re-writing file
  1.08  20-08-1997  Uses the v1.06 kernel, (multi-handles), but has streamlined file handling
                    (less byte tracking, improved method of file pointer seeking).
                    Overall speed improvment approx 1%.
  1.09  20-08-1997  Combines the improvements of 1.07 and 1.08
  1.30  04-09-1997  Fastest file handing, but also has other improvments:
                   -m, -u and -t switches now implemented (as in old GR, but more reliable)
                   -' switch added (converts ' on command line to ")
                   -x switch added (to be used with -d, or -d and -m/-u) and an & in <new>
                   -S Switch - Borland compiler bug workaround:
                   If arguments on command line ended with a backslash ("\") but the argument
                   was in quotation marks, the program would receive the wrong information.
                   V1.30 has a work around, which, for any reason
                   can be disabled with a -S switch.
  1.33  ?          ??
  1.34  ?          Streamlined version of 1.33
  1.35  ?          Bugfixes
  1.36  ?          One minor bugfix
  1.40   ?         First coding towards a multi-file version
  1.41   ?         Command line interpreter strealined saving about 550 bytes
                   of EXE code.
  1.43  22-04-1998  Fully functional version   EXE size: 31025
  1.44  24-10-1998  Turbo Enhanced
  1.50A 06-08-1999  GR tested & finished,[TURBO] EXE size: 30785 Bytes when built: D=0, L=0, W=0, T=1
                   (GR1.50A contains only comsetic improvements over GR1.44)
  1.51  07-01-2001  No bug fixes, but added the facility to be compiled for OS2
                   [TURBO] EXE size for DOS remains at 30785 and 27439 for LITE
  1.52  24-01-2001  Minor bugfix improvement to /N: switch - if <new> is zero length, -N now still works
                   /N:  will now report an error, as will /N:{and another arg}
                    Minor bugfix for when /G and /E used - text arg case preserved
                   Also, if you use "/GE Path" this will now match with an env string call
                   PATH or Path. If that env string doesnt exist, the text arg used will be Path
  1.53  25-01-2001  Simplified WINVER statement to make it easier to compile for different platforms
  1.54  28-01-2001  -K switch added
  1.55  09-02-2001  Command line interpreter rigorously debugged...replaced if switch==K with while switch==k
  1.55A 09-02-2001  Slightly reduced code size by using findlen instead of strlen(find)
  1.56  18-04-2001  Switches[] modified to be compatible with GRX
  1.57  19-04-2001  Minor command line int bug fix
                    Improved help for /N command
                    Slight bugfix on /K command to prevent a 6th instance possibly causing program corruption before being handled
                    Another CLI bugfix to do with the end of the for loop and breaking on errors
                   General switch processor sped up
                   /K and /L (to be implemented soon) now only work in the full version, not the lite version
                   Now use randomize() function instead of srand() for /K
                   Issue considered: GR S.TXT "fi@st" "before&after"
                   If "FIRST" exists, it gives: "beforeFIRSTafter"... should "first" be in lower case? Verdict: No change -
                   consistent with original GR - and how would GR know which case to use?
                       Issue considered: Now if you use /t or /b: GR S.TXT "hor@e" & -t
                       At the moment GR would insert/append: "hor@e", (upon no match) which might not be desirable
                       Suggest: if /t or /b commands used, with & (so no right hand padding will never
                       be needed because the new string will automatically be equal or larger and the search string
                       then a present /n: switch could be used to change the @ to the n:xx character
                       ADDED - to v157
  1.58  20-04-2001  /W implemented on the above issue - instead of using /N use /W just in case you wanted /N for something else
                   /N and /W CLI revised and improved - more intelligent:
                   /N is okay
                   /N: will give error (as before)
                   /NT bugfixed - will reliably accept /N and then pass /T (or whatever) on as another argument
                   /N:291A will pass on: /N:29, plus /1 as a switch (correct), and /A as a switch
                   /N:592 will pass on: /N:59 and /2 as a valid switch
                   /N:295 will pass on: /N:29 and an error because /5 isnt a valid switch
                   /N:023 will pass on: /N:0 and /2 and /3 as switches
                   Note: /N and /W tests possible further numeric args to see if they are valid (1-4). Text args are validated by the rest of the
routine
  1.59  24-04-2001  Help screen split up. /? and /H shows basic information. /F shows further information.
  1.59B 25-04-2001  Slight problem with WINVER fixed - Copyright message now shows correct OS because WINVER should be 0,2,3 as settings not
0,1,2
  1.59C 26-04-2001  Another WINVER problem fixed to do with the ctrlbreak handler
  1.59D 28-04-2001  Fix to the /K checker - new[0] replaced with neww[0]
  1.59E 03-05-2001  No changes. Version number changed to stay in line with GRX.
                   Code checked against BCW 5.02 compiler - OK
  1.59F 16-05-2001  Revision to the code which fixes "arg with space\" problems on command line
                   /Z still disables this - but /Z must be present on the command line before the arg with the problem
  1.59F2            1.59F - MK2 revised with a method that can run recursively to deal with multiple instances of the problem in one arrays[]
  1.60  18-05-2001 to
       04-06-2001  First version of GR able to handle files >64K
                   Included in this, is a modified WINVER option:
                   0 - DOS : 1 - DOS > 64K : 2 - WIN : 3 - WIN > 64K : 4 - OS2
                   MAXBUFSIZE for 64K only programs (0,2) is now set to 65520 not 65530
                   MAXBUFSIZE for other programs is 2147483631 (max size of a signed long - 16 bytes for segment area)
                   Please note: Recommended memory model is still COMPACT
                   Debug info in OBJs can be turned off to reduce EXE size
                   Fast Huge pointers should be disabled for the large DOS and Windows versions
                   OS2 isnt treated as a large version of GR - since it works perfectly in standard mode in that platform
                   GFAR is defined to cope with requirements for different platforms:
                   GFAR = 0   no >64k activity going on
                       (long ints used anyway where necessary ... most ints have to negative at some point and be able to reach 65520)

                   GFAR = 1   >64k activity in use. No special pointer work needed. (For OS2/Linux)
                   GFAR = 2   >64k activity in use. Huge pointers are needed. Fread and Fwrite functions are
                              modified to copy with 65520 chunk limits
  1.61  09-06-2001  >64K support now fully working, still needs further testing. >64K programs now have "+" in version description
                   Following variables renamed: string to strng, handle to hndle
                   Fseek(), Fwrite() and Fread() statements are now checked in DOS & WIN >64K versions
                   so that these functions never operate on more than 65520 bytes at a time
  1.62  11-06-2001  Some code optimizations to reduce code size and slightly increase execution speed:
                   Replace arrays[a][b] with currbyte - a single char to prevent exessive referencing
                   Replace strlen(arrays[a]) with arrayslen - an int
  1.63  12-06-2001  putext() now completely removed from the program if lite==1
  Copyright message moved to slightly ealier part of the program so that
                   User gets something on screen before file is processed
  1.64  28-07-2001  Code symmetry improved with GRX and whitespace tidied
                   mltfix() remains as an unused unneeded fix to a suspicious pointer conversion in search()
                   mltfix() declarations remain in code but completely disabled
  1.64N 30-07-2001  GR and GRX reincorportated into one source file. Use PROGFUNC to determine if you want to compile for GR or GRX
  1.65  31-07-2001  Writeblock now returns short int. writeback() corrected to suite - this has made writeback() faster since fewer 4 byte ints
(long) have to be moved around
                   and Writebytes modified - newwlen now used, b and d now short ints
       02-08-2001  Speed
                   Slight bugfix - CONIO now auto reincluded when DEBUG==1 for getch() function
  1.66  01-08-2001  Failed attempt at improving speed. Cause of poor performance found to be compiler options.
                   Recompiled GR165 outperforms the GR166 with the "FILESYNC disable" feature.
                   GR165 remains the current version.
  1.67  04-08-2001  Feature added: /L (as first discussed on version 1.57)
  1.67B 06-08-2001  Alternate method of /L used. This uses an offset to force search() for start after the last
                   rejected match. Search still returns searchr as a pointer from midbuf, not from midbuf+offset
                   Faster, more efficient, and smaller code, plus correct match reporting
                   this alternate method makes you wonder why we bothered with the first method in the first place!
  1.67C 06-08-2001  GR now reports back to the command line the number of ignored matches
       to          Also, you can now use /L0 to force GR to ignore all matches
       11-08-2001  /L code now completely removed on lite version
                   Revised help for /L command
                   Fixed command line interpreter bug that was introduced on v164
                   On GR (not GRX), bad command line info might not be reported
                   Maximum instances of /K (kswitch/kmode/kmatch) increased from 5 to 10
  1.67D 14-08-2001  /L Command line interpreter (CLI) modified to work as /L:
                   /K:mC argument length better checked
                   Command line interpreter sped up for failing args
```

```
                  Revised shutdown of loadfile func. If any problem, the file is definitely closed.
 1.67E 14-08-2001  -B:C and -T:C implemented to create file if one doesnt exist
 1.67F 20-08-2001  -B:C and -T:C :   Now works in in GRX (for single filename use). Also reports filename back to screen when creating new file
                  which may be useful if you were using GRX with a single filename
                  Small bugix to the /l CLI
 1.68  20-08-2001  Renameable special characters: -n[:b]
                  Program execution sped up as a result. Because special chars. are now stored
                  as a variable, spec. cars (sc) can be disabled by turning the relevant speccar[]
                  to a null terminator. However, this method has only be used fully in search() to improve speed
                  CLI more bugfixed....made sure that /K: and /N: etc without arg will report error
 1.68C 08-10-2001  Heavily revised CLI MK1 - always the achilees heel of GR
1.68C2 10-10-2001  MK2 - tested and for now appears to work. Now for code size reduction
1.68C3 10-10-2001  MK3 - slight code size reduction - non / and - args now dont have argsmet checking - no need
                  Major fix (big problem, small code) which now makes sure GR reports error if there are more than 3 text args on command line
                  This problem was a remnant from old CLI
1.68C4 11-10-2001  New CLI has had code reduced
                  Fixed problem (introduced in 1.68C) where /F would only bring up standard help screen.
                  Slightly revised in built help
 1.69  12-10-2001  Fixed problem where /K action was disabled because of a previous code reshuffle (in v.167?)
1.69A  18-10-2001  Added functionality to CLI - switches can be seperated by : if the user wants
1.69B  19-10-2001  Small bugfix to pointerfix() for OS2 support
1.69C  27-10-2001  Additional functions - ability to write changes to a different file - /$
                  The user can specify CON as the output file
                  For GRX, this name is the default output directory
                  Remove putext from the program as this is no longer appropriate
                  Bugfixed special characters - & not working properly
                  Bugfix to CLI where a command line with 2 text args but no switches would give an error
                  (re-implemented argsmet and argsokay checking for text args)
                  Added enhancement to search() which reduces code size and improves execution.
                  Bugfix to main - newwlen and findlen now calculated correctly
1.69B2 01-11-2001  This is GR169B with the bugfixes from 169C-Beta, but with no function enhancements. Putext is not removed.
1.69C  02-11-2001  Revised 169C beta to standard release. Added /~ switch to disable intelligent file writing
       to         Updated writeback() diagnostics. Checked /GE operation for /$ switch - OK
       03-11-2001  Updated inbuilt help. Checked: GR cannot make lower case filenames/dirs, so there is not point
                  in the CLI preserving cases. Updated error code reporting
                  Now reports error if user gives /$ but doesnt give a text arg
1.69D  03-11-2001  Renamed /L to /#
                  /#:0 now reports back possible matches to DOS (GR only)
                  Added MAXSTRNGSIZE to have more flexible control over string size limits
                  Added MAXSRSIZE for future /L load search/replace from file option
1.69E  04-11-2001  GRX now displays a total of skipped matches. Plus, returns count of /#:0 to DOS for GRX
                  Pre-start DEBUG added to check inital memory allocation
1.69F  05-11-2001  Revised writebytes() function which no longer needs strng to be 256 bytes or the size of find
                  Instead, it uses a new global ptr, tmpptr
                  Added /&[:x] switch (although lite versions can only use /&)
                  Reduced size of *strng to MAXSTRNGSIZE instead of MAXENVSIZE
                  Bugfix to the new CLI - better error checking
                  Improved pre-start debug - you can now press ESC to abort test run
1.70A  05-11-2001  Recursive directory support added for GRX
       RC1  to            -Updated /$, /B and /T switch for GRX
      27-01-2002  -GRX: If no work is done, any output directory from /$ is removed
                  -getfilepath() updated AND bugfixed. Bugfix applied to GR169B2>Gr169B3 and GR169F>GR169G
                  -Fixed ininame initialisation that was overwritting a byte it shouldnt
                  -Tidied /B:C and /T:C support to regulate startbuf
                  -Fixed where the program might try and close fwptr even if its not open
                  (this shouldnt have been a problem but has been done for good practice)
                  -Improved program stability if a disk error is encountered by writeback() - the file
                  will not be truncated if there is an error condition
                  -GRX: Discovered that 1000 or so uses of malloc() by program caused heap corruption
                  This is a compiler or OS issue. Workaround: GRX now starts with a buffer designated by GRXMINBUFSIZE
                  If more space is needed than is provided by this buffer, a new one if mallocated but this buffer
                  is then kept for subsequent files. More reliable...and much faster.
                  -Minor cosmetic memory demallocation bug fixed - applied to GR169G>H and GR169B3>B4
                  -Modified WINVER to stop unnecessary far pointers being used for pointerfix()
                  -Changed copyright message
                  -For OS2, WINVER is now 9. WINVER=4 and 5 added for Win32 support
                  -Win32 working (compile for command line) including long filenames and recursive directory support
                  -Added /P switch to force a keypress for GR/GRX after the program has done its run
                  -Added /$:f and /$:a switch for GRX, /$:a switch for GR (although using :F wont cause error for GR)
                  -Implemented /v - using this prevents GR/GRX from showing verbose information
                  -/B and /T will now create a new file if the old file size is zero bytes
                  (gr wont normally work on/load into memory files of zero bytes)
                  -For /R and /$ - introduced directory cyclic operation check
1.69G  08-11-2001  Getfilepath bugfixed...this has been done so that there is a fully working version of the 169 series
1.69B3 08-11-2001  Getfilepath bugfixed...this has been done so that there is a reliable version of the program for Stephen's use
1.69H  20-11-2001  Minor cosmetic memory demallocation bug fixed - applied to GR169G>H and GR169B3>B4
 and 1.69B4        Modified WINVER to stop unnecessary far pointers being used for pointerfix()
1.69I  18-01-2002  Changed copyright message
 and 1.69B5
1.70A  29-01-2002  1.70A done & finished, thats it for this version!
 -- 1.70A remains the latest version --
1.70B  10-02-2002  Revised copyright protecion, cname
1.70C  07-03-2002  GRX now displays a warning message if it is unable to go deep enough into the directory structure
1.70D  08-03-2002  Implemented /L (not available in LITE versions)
1.70D2 10-03-2002  Corrected a spelling mistake in basic help thats been there since V159
                  Program execution sped up by writenew() no longer using copies of newwlen and findlen
                  These copies were long ints, program now using unsigned ints saving an extra four bytes being moved around
                  Fixed a whopper bug where find's length was governed by MAXSTRNGSIZE not MAXENVSIZE
                  Slightly improved DEBUG==1 startup
                  Fixed /J not working for GRX (this code needed to be revised anyway for /L)
```

```
                   Revised randgen to remove /k:6 (add cr) from it, this will now be done by writebytes in a future version
                   and therefore changed randgen not to return errors
                   Replaced strcpy with movemem for all find/neww actions (namely copying between neww and backupneww)
 1.70D3 27-03-2002  Further developed /L support by getting newwlen and findlen working properly as unsigned ints
       to          Reduced code size and increased speed on writenew and writebytes by using less long ints
       04-04-2002  Started /K:6 re-adding support by introducing kcrswitch. all kswitch and kmatch now sorted as they are stored
                   so that all requests for /k:6x are stored in kmatch[0] to kcrswitch, to slightly improve performance
                   More performance considerations: now that neww can be up to 64k, use switches[10] to possibly disabled checking for &
                   so that when we have found once that neww doesnt have & in it, its not checked again
                    (this is speccar[2])
 1.69B6, 1.69J  04-04-2002  Last versions of these, with the following improvements from 170D2 and D3 added:
                   Spelling mistake in help corrected, fixed /J not working in GRX (not fixable in 169B5), speccar[2] performance improvement
added
                   Fixed the whopper, but not serious bug where find's length was governed by MAXSTRNGSIZE not MAXENVSIZE (not needing to be
fixed in 169B5)
  1.71  05-04-2002  /L implementation fixed and tested, /k:6x reimplementation fixed
                   This had to include making kcrswitch/kmode etc global
                   Tiny but serious writeback() bug fixed - this was introduced in 1.70A
  1.72  20-04-2002  Disabled /R for LITE vesion of GRX (should have already been done)
                   Did this by setting up alternative swmatch for progfunc1 lite and non lite versions
  1.73  10-06-2002  WINVER renamed to more accurate PROGTYPE
  1.74  25-01-2003  Date change only version; 1.72/1.73 had not been released, so 1.74 has been released with new year
  1.75  11-08-2003  Compilation option added for further future OS Support
 1.76-1 12-12-2003  Added option for OS2 to be compiled for small mode (in case needed)
 1.77-1 29-01-2006  Minor code tidy to try and kill some bugs
 1.77   13-08-2006  Major bugfix on search function reporting erroneous matches
 1.78   19-08-2006  Fixed /q /f causes infinite loop - swmatch was corrupted in tests from 176 onwards
                   Fixed minor CLI issue that caused /K with no extra parameters to infinite loop
 1.79   20-08-2006  Fix for gr file.txt "@@ " causing hang
 1.79B  20-08-2006  Improved coding so that specifying gr file.txt "@" or gr file.txt "@@" actually works
 1.79C  16-10-2006  Minor changes to global string declarations, increasing by 1 character. 1st compile for OpenWatcom
                   Thanks to Stephen for his assistance and testing with OS2 and WATCOM
 1.79D  17-12-2006  Fixed gr179c #0 "," /i not working - code was finding extra matches
                   Updated command line help with note on & switch
                   Added WATCOMC define to compile for Watcom
                   Updated maximum large file size to 2147483583 from 2147483631
 1.79E  17-12-2006  Add /$:B for GRX - makes the /$ output to a directory append to existing files
                   Change max number of command line args from 16 to 128
 1.79F  05-01-2007  Added command line feedback help
                   Includes better help for text args and unrecognised args
                   Automatically added Lite description to program description
 1.79G  06-01-2007  Added checking for creation of temp file - just more robust
 1.79H  05-05-2007  Minor bugfix on Error reporting with /$ <file> when <file> in use x=-6
                   Minor codetidy on var tempfcreated on progfunc=0 (not needed)
                   Added -P on help for GRX as this was missing
 1.79I  06-05-2007  Consolidated help -F now available at -H:2
 1.79J  06-05-2007  /V was removing extra information - not working correctly, reprogrammed:
                   /V:0 is now silent mode, V:1 is quiet, /V:2 is standard, /V:3 is verbose
                   (removed /q)
                   /V:3 displays more writeback information
                   /L error reporting improved especially for when no filename provided
 1.79K  06-05-2007  makealtfnamedir now can create multiple directory levels
                   STRUPR filenames for /L find and neww
                   Minor update to % speccar processing for find
 1.79L  09-05-2007  Some fixes to the CLI to prevent buffer overruns (STRNCPY instead of STRCPY)
 1.79M  22-07-2008  Fixed loop on GR179L ..\history " %%i" " %%i" /t /m /$:a ..\result.csv (and /b instead of /t)
                   Problem caused by use of /t or /b function on Win32 compiled version
                   caused by unsigned/signed int mistake in writenew()
                   Fixed additional bug in writebytes() that could have crashed on /L:files more than 32k
                   Also fixed LITE==1 failure on debug
 1.79P  23-07-2008  Fixes to searchtwo() to use long ints permanently
 1.79Q  07-10-2008  Added compatibility for MS Visual Studio.NET 2003 C++
        to          Renamed WATCOM to COCOMPAT
       01-11-2008  Fixed bug in aswrite (for GFAR==2 - DOS and Win16 large mode)
                   pointer was not being modified correctly to write subsequent 64k blocks
                   Code revisions to better support /l: for files above 64k (neww and find)
                   Including changes and revision to 16bit support
                   For GFAR==2 to make sure all fwrites are done by aswrite
                   This has also meant at the addition of aswriteb for GFAR==2
                   To enable p_ptr to be redirected to another copy of fwrite
                   That will work on chunks over 64k
1.79R-1 01-11-2008  Replace fwrite error checking with ferror
1.79R-2 02-11-2008  Debug for OS2 and Watcom
                   Added defines for GRX and Watcom for findfirst and findnext
 1.79R  05-11-2008  Final - with one small improvement to writeback()
 1.79S  06-11-2008  Bug in gr <name> <search> <replace> -d -m (combinination of d and m switches)
                   When the first match was searchr==0
 1.79T  07-11-2008  Finally turned on >64K support for /L for Full Win32 only
                   It may be possible to turn it on for Win16 earlier, but this will require neww and find to be char *huge
                   Couple of minor bug fixes
 1.79U  08-11-2008  Improvements and fixes to writebytes related to & searchtwo calls
                   Added stcounts for verbose information to show the number of times & has been used
                   Add /K:Hs H random lower case Hex (kmode 72)
                   Fixed CLI loop on /K handling when the mode byte was not valid
                   Changed help description for -P switch for GR and GRX
1.79V-R1 22-11-2008 Add /K:Is I random upper case Hex (kmode 73)
                   Improved compatibility for GRX and Watcom - not complete
                    Improved compatibility for GRX and MS2003 - not complete
1.79V-R2 22-11-2008 Added fix that stcounts is not cumulative for GRX
1.79V-R3 24-11-2008 Tweaks
```

```
                   Added compiler time checks to add a Printf warning statement
                   to warn a user of the program if it does not appear to be have been
                   compiled on the target compiler as set by COCOMPAT
                   Fixes to rcsubdirmgr to carry on properly when maximum sub dir length found
                   and to also not incorrectly pass an error if an empty dir was the last item processed
                   (was meaning that Error no files to process being reported)
                   Fixes to Short file versions displaying %u or %l instead of %d
1.79V-R4 27-11-2008 Change file name handling string size to 261 from 256
                   Enabled ctrlbreak handler for LITE version
                   Installed CTRLBREAK handler for Watcom
1.79V-R5 25-12-2008 Re-disabled CBREAK Handler for OS2 and WIN32 (all compilers)
                   Watcom compilers Break handler only now for DOS
                   Watcom and DOS-  Link or Make library path needs to be added to compiler options for DOS
1.79V-R6 27-12-2008 More Watcom porting for GRX for the return values for findfirst and findnext
                   (which are translated to _dos_findfirst etc) being non-zero instead of -1 on error
1.79V-R7 28-12-2008 Added GR display message to incorporate compiler info
                   More porting work for GRC for MS2003
 1.79V  29-12-2008 Summary of work from 1.79U to 1.79V
        to         Added /K:Is for random upper case (hex)
       01-01-2009 Fixes to rcsubdirmgr for directory handling
                   Fix to stcounts should not be cumulative for GRX
                   Added DOS ctrlbreak handler for Watcom and enabled for Lite
                   Added warning statements displayed at program run if the compiler type doesnt seem to be right
                   Added small compiler display string on first help page
                   Some other minor bug fixes and tweaks
                   Completed porting program to WatcomC and Visual Studio 2003
                   Added code compiler compatibility table
                   Added Nagware option
                   Added third help screen with contact information
                   Confirmed CLI pointerfix for "fgfgf [backslash]" still required for watcom and ms2003 (-z switch)
                   Added second cnameb protection
 1.79W  04-01-2009 Revised help text for V switch
 1.79X  04-05-2009 Fixed /e /g <enivir arg> not working
       to          Added some more debug info
       03-08-2009 Removed extra bracket from first help page
                   Tidied and made more consistent the help
                   Added a fix for GR <file> <text> /d
                   The empty line - extra CR was not being removed. Added /D:P switch to turn off
                   this automatic removal and Preserve the line
                   Note:/D:P is switches[40] in 1.79X and switches[49] in 1.80
 1.79Y 05-08-2009  Revised loadfile to be able to report difference between read only files, and file not found
                   This needed error code -30 and errorlevel to DOS of 9 for GR
                   Revised layout of DOS return codes in help screen to add new code
                   Changed the /K code which checks to make sure the same byte isnt being used as a special character
                   Now it will only report bad command info on matching the only speccar used in <new>, speccar[2] - &
                   Added BETA support for Win64 compilation. Needs Visual C++ 2005
                   Modes are Progtype=6 2GB max, and Protype-7 8GB max (huge mode)
                   Removed /J on GRX and replaced with /K:RM for GR/GRX and /K:RF GRX only
                   Reworked and made faster and more reliable randgen
                   Changed output of GRX errorcode -4 to -6 by final main exit change to make it
                   match GR
1.79Y5 25-12-2009  Fixed /N and /W cant use an ASCII code of 0 - single source bug
1.79Y6 12-04-2010  Checked /N and /W operation OK - ASCII code cannot include leading zeros
                   Fixed loadfile opening file RW when should be RO and vice versa
                   Caused by mis-handling /# switch
1.79Z  15-04-2010  Added /#:i switch to ignore first match
                   Added /#:j switch to ignore last match only
                   Fix /#:L setting disappearing after first file on GRX
1.79Z2 09-06-2010  Revised integrated help text for /# command
1.79Z3 13-07-2010  Debug GR file /l file2 "&" /D where file2 has multiple lines and match found
                   Found cause to be search() looking for next carriage returns
                   Should have been search from findlen onwards to avoid hitting
                   Carriage returns in found pattern
                   Fixed /D:P not working
1.79Z4 17-07-2010  Added /D:L to preserve LH contents of line, /D:R to preserve right
1.79Z5 19-09-2010  Changed speccar 4 from percent to ! symbol
                   Added /H:V to display version information only
1.79Z6 24-09-2010  Minor tweak to Lite version variable list to cure compiler warning
                   Prevented Lite message appearing on /H:V on Lite version
1.79Z7 30-10-2010  Fixes to the search engine to make sure that "#@" and other functions
                   in search are used when search string just made up of wildcards
                   Also fixed GR not returning zero matches on error with /#:0
                   Fix to writenew to write text from midbuf instead of find
1.79Z8 06-12-2010  Added ability to add /#:<number> as any number from 1 to 999999999
                   instead of just 1 to 9
                   Fixed reporting matches not working on GRX
                   Added /#:R to report matches unless error for GR and GRX
1.79Z9 08-12-2010  Updated help text for /#:R above
                   Added support for up to ten /#:number, either on separate /# commands or separated by commas
                   Solved /' switch not working after first file or match, when /k with random features used and /k:rm or /k:rf
                   Reliability enhancement to asnewwfwrite
1.79ZA 10-12-2010  Added support to seperate /K arguments with commas
1.79ZB 18-12-2010  Added option for /k: modes for percent, greater and less than symbols
                   Performance enhancements to randgen()
1.79ZC 27-02-2011  Fixed /#:0 and /#:R core functionality not working since Z8
1.79ZD 17-06-2011  Solved /t:c and /b:c string not processed for /K:6x switches - kcrmode
                   This was caused by a problem in writenew, which used its own handler for writing the tag string
                   instead of using writebytes with a function redirection if kcrswitch being used
```