Digital Research tex 2.1

This is a c-port of Digital Research's tex version 2.1. It is based on my PL/M compilation of the original, along with fixes I had previously applied. The PL/M source for this and other versions can be found at

https://github.com/ogdenpm/intel80tools/tree/master/cpmsrc/tex

The port now works on Windows and Linux, although the kbhit has been stubbed out as both operating systems support control C to interrupt a program. Additionally paging now only switches to raw input mode when needed, so processing of console input as part of normal tex processing should work. Note for Linux, to prevent the options e.g. $P from being expanded by the shell, prefix the $ with a \.

On Windows the logic to determine if output is to a device, tests for the standard windows devices, on Linux it checks for a /dev/ prefix.

In this directory you will also find a copy of the application manual tex.pdf and the notes below show additional command line options and dot commands supported by version 2.1 and the changes I made as a result of the port.

Additional Dot commands:

The notes below reflect my best guess at the additional dot commands, from looking at the original decompiled source code. A number are options that require a printer, probably Diabalo and Qume, for handling various features including proportional spacing.

If you have one of these printers, please let me know if I have made an error.

```
.BO n   Bold for next n characters
.C2 nc  Set literal lead in Char to c (n is not used)
.CC nc  Set command lead in char to c (n is not used)
.CL txt Process txt as non file related command line options
.CS n   Set character pacing
.EC nc  Set escape char to c (n is not used)
.FI     Set formatting to add extra space after . ? !
.FM n   Set footer margin
.FT txt Set footer text
.IX txt Write index line
.NE n   New page if not at least n lines left
.NF     Single space after . ? !
.NX txt Next file. See notes below
.PC nc  Set page number marker (default is %)
.PS     Enable proportional spacing
.QB     Quit bold / shadow
.QS     Quit bold / shadow
.QP     Quit proportional spacing
.QU     Quit underlining
.RD     Get user input from response file / command line
.RF txt Set response file
.SH n   Set shadow printing
.SO txt Source from file
```

```
.ST    Wait to continue
.TA ... Set tab stops
.TC txt Write TOC line
.TM txt Write message
.UL n   Start Underline
```

Command line options

To support new features Tex 2.1 supports a number of command line options.

```
usage:
tex [-v|-V] | inputfile+ texOption*
Note the options start with $ and are after the input files

If -v or -V is used, they must be the only option. They provide version information

The supported tex options are
$S          Paging or manual feed for non file output (modified for port)
$F          Use form feed
$D          Diabalo printer?
$I nn [mm]  Range to print nn-mm (mm defaults to all)
$N nn       Initialise page number to nn
$R txt      Set Response file
$Q          Qume printer?
$C          Prompt for continue on change of input file
$T path     Path to tex file (modified for port)
$P path     Path to output (modified for port)
$EY         Enable error printing
$EX         Disable error printing (default)
$X path     Path to index file (modified for port)
```

Handling of file options

In the original Tex v2.1 the file options only took a drive letter A-P for disk files. Dependent on the $ option other letters were used: X for the console, Y for the LST defined and Z for no file, all others were treated as errors.

With hierarchical directory structures this is restrictive. To work around this the port modifies these as follows

Option	Behaviour
$T path	The path is read from the command line, to the argument boundary. The trailingdirectory separator is optional but will be added if needed

Option	Behaviour
$P path	The path is read from the command line, to the argument boundary. If the path is a dash then the output will be put to the console, which can be paged dependent on other options If the path is a device name, then it is treated as a listing device. Be aware that if this device is CON: then progress messages will appear in line, use the dash variant instead. If it ends in a directory separator, it will be used as the directory where the output file is placed. The file name will be derived from the first input file, with a .prn extent. Otherwise the path is used as the name of the output file, with .prn added if there is no extent. If there is no $P option then a file will be generated based on the input file with the extent .prn
$X path	The path is read from the command line, to the argument boundary. If the path is a dash then no index file will be generated If it ends in a directory separator, it will be used as the directory where the output file is placed. The file name will be derived from the first input file, with a .ix extent. Otherwise the path is used as the name of the output file, with .prn added if there is no extent. If there is no $X option then a file will be generated based on the input file with the extent .ix Using a device name is treated as an invalid option

Other than the $T, $P and $X options, any file names are parsed as per CP/M rules in relation to valid characters.

```
This means that specifically the characters "\n\t ?*=:!=|_[]" are not supported
```

Although this allows / or \ to be included, be aware any name in the tex source files will have the $T path prefixed.

One potential issue is around the $R option. If this is specified on the command line before $T path, then it is will be treated as the file name, relative to the current directory. If after the $T path then it will be prefixed with the $T path.

By changing to the directory containing the tex source files, it is unlikely that the $T will be necessary. Also unless overriding the default names or supressing the ix output, the $P and $X options are also unlikely to be used.

```
Mark Ogden 3-May-2023
```