

- [JanetTerra](#) Apr 16, 2006

It is possible to load and display graphics in formats other than bitmap in Liberty BASIC programs. This does require the use of the atl DLL or a third party DLL such as [lbbrowse3.dll](#). Finding the dimensions of a [Portable Network Graphics](#) (PNG) file is very similar to that of getting the dimensions of a [bitmap file](#). The file must be opened and the first three Lines Inputted. The 7th, 8th, 11th and 12th Bytes of the third Line Input are then parsed for the pertinent information.

```
' Choose a PNG File
  Filedialog "Finding the PNG Dimensions", "*.*", pngFile$
  If pngFile$ = "" Then End

' Obtain pngInfo in 3 Line Inputs
  Dim pngInfo$(3)
  Open pngFile$ for Input as #png
    For i = 1 to 3
      Line Input #png, pngInfo$
      pngInfo$(i) = pngInfo$
    Next i
  Close #png
  Print

' 1st Byte is a High Bit to reduce possibility text file is mistaken for or png file
  hiBitDec = Asc(Mid$(pngInfo$(1), 1, 1))
  hiBitHex$ = DecHex$(hiBitDec)
  Print hiBitHex$ 'hiBitHex$ = "89"

' Next 3 Bytes are the ASCII values of P N G for easy identification of format
  For i = 2 to 4
    Print Mid$(pngInfo$(1), i, 1); " "; ' Prints P N G
  Next i
  Print: Print

' Next are Bytes coded to detect the line ending conversion
' 0A = Line Feed
' 0D 0A = Carriage Return plus Line Feed
' 1A = Stop Display of File
  Print pngInfo$(2)
  Print

' The rest of the information follows in the third Line Input
```

```
' Each block of information is referred to as a chunk
' The first chunk in this block is always the header identified by IHD
R
' Only a few of the Bytes need to be parsed to find the width and height
For i = 1 to 4
    Print Mid$(pngInfo$(3), i, 1); " "; ' Prints I H D R
Next i
Print: Print
' The PNG Width is stored in the 7th and 8th Bytes of this third Line Input
pngWidth = Asc(Mid$(pngInfo$(3), 8, 1)) + Asc(Mid$(pngInfo$(3), 7, 1)) * 256
Print "pngWidth = ";pngWidth
Print
' The PNG Height is stored in the 11th and 12th Bytes of this third Line Input
pngHeight = Asc(Mid$(pngInfo$(3), 12, 1)) + Asc(Mid$(pngInfo$(3), 11, 1)) * 256
Print "pngHeight = ";pngHeight
Print
End
```

To use this within your Liberty BASIC program, you only need the these lines

```
Open pngFile$ for Input as #png
    Line Input #png, pngSignature$
    Line Input #png, pngLineEnd$
    Line Input #png, pngInfo$
Close #png
pngWidth = Asc(Mid$(pngInfo$, 8, 1)) + Asc(Mid$(pngInfo$, 7, 1)) * 256
pngHeight = Asc(Mid$(pngInfo$, 12, 1)) + Asc(Mid$(pngInfo$, 11, 1)) * 256
```