# PGM Format Specification

**pgm**

Updated: 03 October 2003
Table Of Contents

## NAME

pgm - Netpbm grayscale image format

## DESCRIPTION

This program is part of Netpbm.

The PGM format is a lowest common denominator grayscale file format. It is designed to be extremely easy to learn and write programs for. (It's so simple that most people will simply reverse engineer it because it's easier than reading this specification).

A PGM image represents a grayscale graphic image. There are many psueudo-PGM formats in use where everything is as specified herein except for the meaning of individual pixel values. For most purposes, a PGM image can just be thought of an array of arbitrary integers, and all the programs in the world that think they're processing a grayscale image can easily be tricked into processing something else.

The name "PGM" is an acronym derived from "Portable Gray Map."

One official variant of PGM is the transparency mask. A transparency mask in Netpbm is represented by a PGM image, except that in place of pixel intensities, there are opaqueness values. See below.

The format definition is as follows. You can use the libnetpbm C subroutine library to conveniently and accurately read and interpret the format.

A PGM file consists of a sequence of one or more PGM images. There are no data, delimiters, or padding before, after, or between images.

Each PGM image consists of the following:

1 A "magic number" for identifying the file type. A pgm image's magic number is the two characters "P5".

2 Whitespace (blanks, TABs, CRs, LFs).

3 A width, formatted as ASCII characters in decimal.

4 Whitespace.

5 A height, again in ASCII decimal.

6 Whitespace.

7 The maximum gray value (Maxval), again in ASCII decimal. Must be less than 65536, and more than zero.

8 Newline or other single whitespace character.

9 A raster of Height rows, in order from top to bottom. Each row consists of Width gray values, in order from left to right. Each gray value is a number from 0 through Maxval, with 0 being black and Maxval being white. Each gray value is represented in pure binary by either 1 or 2 bytes. If the Maxval is less than 256, it is 1 byte. Otherwise, it is 2 bytes. The most significant byte is first.

A row of an image is horizontal. A column is vertical. The pixels in the image are square and contiguous.

10 Each gray value is a number proportional to the intensity of the pixel, adjusted by the ITU-R Recommendation BT.709 gamma transfer function. (That transfer function specifies a gamma number of 2.2 and has a linear section for small intensities). A value of zero is therefore black. A value of Maxval represents CIE D65 white and the most intense value in the image and any other image to which the image might be compared.

11 Note that a common variation on the PGM format is to have the gray value be "linear," i.e. as specified above except without the gamma adjustment. **pnmgamma** takes such a PGM variant as input and produces a true PGM as output.

12 In the transparency mask variation on PGM, the value represents opaqueness. It is proportional to the fraction of intensity of a pixel that would show in place of an underlying pixel. So what normally means white represents total opaqueness and what normally means black represents total transparency. In between, you would compute the intensity of a composite pixel of an "under" and "over" pixel as under * (1-(alpha/alpha_maxval)) + over * (alpha/alpha_maxval). Note that there is no gamma transfer function in the transparency mask.

13 Characters from a "#" to the next end-of-line, before the maxval line, are comments and are ignored.

Note that you can use **pamdepth** to convert between a the format with 1 byte per gray value and the one with 2 bytes per gray value.

There is actually another version of the PGM format that is fairly rare: "plain" PGM format. The format above, which generally considered the normal one, is known as the "raw" PGM format. See **pbm** for some commentary on how plain and raw formats relate to one another and how to use them.

The difference in the plain format is:

- There is exactly one image in a file.
- The magic number is P2 instead of P5.
- Each pixel in the raster is represented as an ASCII decimal number (of arbitrary size).
- Each pixel in the raster has white space before and after it. There must be at least one character of white space between any two pixels, but there is no maximum.
- No line should be longer than 70 characters.

Here is an example of a small image in the plain PGM format.

```
P2
# feep.pgm
24 7
15
0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
0  3  3  3  3  0  0  7  7  7  7  0  0 11 11 11 11  0  0 15 15 15 15  0
0  3  0  0  0  0  0  7  0  0  0  0  0 11  0  0  0  0  0 15  0  0 15  0
0  3  3  3  0  0  0  7  7  7  0  0  0 11 11 11  0  0  0 15 15 15 15  0
0  3  0  0  0  0  0  7  0  0  0  0  0 11  0  0  0  0  0 15  0  0  0  0
0  3  0  0  0  0  0  7  7  7  7  0  0 11 11 11 11  0  0 15  0  0  0  0
0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
```

There is a newline character at the end of each of these lines.

Programs that read this format should be as lenient as possible, accepting anything that looks remotely like a PGM.

## COMPATIBILITY

Before April 2000, a raw format PGM file could not have a maxval greater than 255. Hence, it could not have more than one byte per sample. Old programs may depend on this.

Before July 2000, there could be at most one image in a PGM file. As a result, most tools to process PGM

files ignore (and don't read) any data after the first image.

**SEE ALSO**

pnm, pbm, ppm, pam, libnetpbm, programs that process PGM,

**AUTHOR**

Copyright (C) 1989, 1991 by Jef Poskanzer.

---

**Table Of Contents**

NAME
DESCRIPTION
COMPATIBILITY
SEE ALSO
AUTHOR