User manual for old pgm functions

pgm Functions

Updated: 22 July 2004
Table Of Contents

NAME

libpgm - libnetpbm functions to read and write PGM image files

SYNOPSIS

#include <pgm.h>

void pgm_init(int **argcP*, char **argv*[]);

gray ** pgm_allocarray(int *cols*, **int** *rows* **);**

gray * pgm_allocrow(int*cols* **);**

void pgm_freearray(gray ****grays*, **int*rows* **);**

**void pgm_freerow(gray ***grayrow*);

void pgm_readpgminit(FILE ***fp*, **int ***colsP*, **int ***rowsP*, **gray ***maxvalP*, **int ***formatP* **);

void pgm_readpgmrow(FILE ***fp*, **gray ***grayrow*, **int *cols*, **gray** *maxval*, **int** *format* **);**

gray ** pgm_readpgm(FILE ***fp*, **int ***colsP*, **int ***rowsP*, **gray ***maxvalP* **);

void pgm_writepgminit(FILE * fp , int *cols*, **int** *rows*, **gray** *maxval*, **int** *forceplain* **);**

void pgm_writepgmrow(FILE ***fp*, **gray ***grayrow*, **int cols, **gray** *maxval*, **int** *forceplain* **);**

void pgm_writepgm(FILE ***fp*, **gray ** *grays*, **int *cols*, **int** *rows*, **gray** *maxval*, **int** *forceplain* **);**

void pgm_writepgm(FILE ***fp*, **gray ****grays*, **int *cols*, **int** *rows*, **gray** *maxval*, **int** *forceplain* **);**

void pgm_nextimage(FILE ***file*, **int * const *eofP*);

void pgm_check(FILE * *file*, const enum pm_check_type *check_type*, **const int** *format*, **const int** *cols*, **const int** *rows*, **const int** *maxval*, **enum pm_check_code * const** *retval*);

typedef ... gray;

#define PGM_MAXMAXVAL ...

#define PGM_OVERALLMAXVAL ...

extern gray pgm_pbmmaxval;

#define PGM_FORMAT ...

#define RPGM_FORMAT ...

#define PGM_TYPE PGM_FORMAT

#define PGM_FORMAT_TYPE(*format***)**

DESCRIPTION

These library functions are part of Netpbm.

TYPES AND CONSTANTS

Each **gray** should contain only the values between **0** and **PGM_OVERALLMAXVAL**. **pgm_pbmmaxval** is the maxval used when a PGM program reads a PBM file. Normally it is 1; however, for some programs, a larger value gives better results.

PGM_OVERALLMAXVAL is the maximum value of a maxval in a PGM file. **PGM_MAXMAXVAL** is the maximum value of a maxval in a PGM file that is compatible with the PGM format as it existed before April 2000. It is also the maximum value of a maxval that results in the minimum possible raster size for a particular image. I.e an image with a maxval higher than **PGM_MAXMAXVAL** cannot be read or generated by old PGM processing programs and requires more file space.

PGM_FORMAT is the format code for a Plain PGM format image file. **RPGM_FORMAT** is the format code for a Raw PGM format image file. **PGM_TYPE** is the format type code for the PGM formats. **PGM_FORMAT_TYPE** is a macro that generates code to compute the format type code of a PBM or PGM format from the format code which is its argument.

INITIALIZATION

pgm_init is identical to **pm_init()**.

MEMORY MANAGEMENT

pgm_allocarray() allocates an array of grays.

pgm_allocrow() allocates a row of the given number of grays.

pgm_freearray() frees the array allocated with **pgm_allocarray()** containing the given number of rows.

pgm_freerow() frees a row of grays allocated with **pgm_allocrow()**.

READING FILES

If a function in this section is called on a PBM format file, it translates the PBM file into a PGM file on the fly and functions as if it were called on the equivalent PGM file. The *format* value returned by **pgm_readpgminit()** is, however, not translated. It represents the actual format of the PBM file.

pgm_readpgminit() reads the header of a PGM file, returning all the information from the header and leaving the file positioned just after the header.

pgm_readpgmrow() reads a row of grays into the *grayrow* array. *format*, *cols*, and *maxval* are the values returned by **pgm_readpgminit()**.

pgm_readpgm() reads an entire PGM image into memory, returning the allocated array as its return value and returning the information from the header as *rows*, *cols*, and *maxval*. This function combines **pgm_readpgminit()**, **pgm_allocarray()**, and **pgm_readpgmrow()**.

pgm_readpgminit() and **pgm_readpgm** abort the program with a message to Standard Error if the PGM image header is not syntactically valid, including if it contains a number too large to be processed using the system's normal data structures (to wit, a number that won't fit in a C 'int').

WRITING FILES

pgm_writepgminit() writes the header for a PGM file and leaves it positioned just after the header.

forceplain is a logical value that tells **pgm_writepgminit()** to write a header for a plain PGM format file, as opposed to a raw PGM format file.

pgm_writepgmrow() writes the row *grayrow* to a PGM file. For meaningful results, *cols*, *maxval*, and *forceplain* must be the same as was used with **pgm_writepgminit()**.

pgm_writepgm() write the header and all data for a PGM image. This function combines **pgm_writepgminit()** and **pgm_writepgmrow()**.

MISCELLANEOUS

pgm_nextimage() positions a PGM input file to the next image in it (so that a subsequent **pgm_readpgminit()** reads its header).

pgm_nextimage() is analogous to **pbm_nextimage()**, but works on PGM and PBM files.

pgm_check() checks for the common file integrity error where the file is the wrong size to contain all the image data.

pgm_check() is analogous to **pbm_check()**, but works on PGM and PBM files.

SEE ALSO

libpbm, **libppm**, **libpnm**

Table Of Contents

NAME
SYNOPSIS
DESCRIPTION
TYPES AND CONSTANTS
INITIALIZATION
MEMORY MANAGEMENT
READING FILES
WRITING FILES
MISCELLANEOUS
SEE ALSO