```
/********************************************************************
            gmap: image structure, greyscale pixels.
The width field contains the width of the image,
- The height field contains the height of the image,
- The raster pointer points to a memory area containing
the image pixel, 8 bits per pixel, 0 is black, 255 is white.
The (0,0) coordinate is the upper left corner of the image, and
its address is raster. The (1,0) coordinate is at address raster + 1.
The (0,1) coordinate is at address raster + width.
**********************************************************************/
typedef struct gmap {
short                width;
short                height;
unsigned char  *raster;
} gmap;

/*********************************************************************
         readpgm()
This function reads the pgmfile whose name is given as argument
and creates a gmap structure filled with the image.
This function allocate the memory requested to store the image.
If either the file cannot be opened or not enough memory
remains readpgm returns NULL
**********************************************************************/
extern gmap *readpgm(char *name);

/*********************************************************************
         writepgm:
This function writes the gmap map into the file whose name is
given as argument.
If the file cannot be opened, writepgm returns NULL.
**********************************************************************/
extern int   writepgm(gmap *map, char *name);
```