NAME

gvgen – generate graphs

SYNOPSIS

gvgen [−**d?**] [**-c**n] [**-C**x,y] [**-g[f]**x,y] [**-G[f]**x,y] [**-h**n] [**-k**n] [**-b**x,y] [**-p**n] [**-s**n] [**-S**n] [**-t**n] [**-T**x,y] [**-w**n] [**-n**prefix] [**-N**name] [**-o**outfile]

DESCRIPTION

gvgen generates a variety of simple, regularly-structured abstract graphs.

OPTIONS

The following options are supported:

−**c** *n*   Generate a cycle with *n* vertices and edges.

−**C** *x,y*   Generate an *x* by *y* cylinder. This will have *x*y* vertices and *2*x*y - y* edges.

−**g** *[f]x,y*
              Generate an *x* by *y* grid. If **f** is given, the grid is folded, with an edge attaching each pair of opposing corner vertices. This will have *x*y* vertices and *2*x*y - y - x* edges if unfolded and *2*x*y - y - x + 2* edges if folded.

−**G** *[f]x,y*
              Generate an *x* by *y* partial grid. If **f** is given, the grid is folded, with an edge attaching each pair of opposing corner vertices. This will have *x*y* vertices.

−**h** *n*   Generate a hypercube of degree *n*. This will have 2^n vertices and $n*2^{(n-1)}$ edges.

−**k** *n*   Generate a complete graph on *n* vertices with *n*(n-1)/2* edges.

−**b** *x,y*   Generate a complete *x* by *y* bipartite graph. This will have *x+y* vertices and *x*y* edges.

−**p** *n*   Generate a path on *n* vertices. This will have *n-1* edges.

−**s** *n*   Generate a star on *n* vertices. This will have *n-1* edges.

−**S** *n*   Generate a Sierpinski graph of order *n*. This will have $3*(3^{(n-1)} - 1)/2$ vertices and 3^n edges.

−**t** *n*   Generate a binary tree of height *n*. This will have $2^{n}-1$ vertices and $2^{n}-2$ edges.

−**T** *x,y*   Generate an *x* by *y* torus. This will have *x*y* vertices and *2*x*y* edges.

−**w** *n*   Generate a path on *n* vertices. This will have *n-1* edges.

−**n** *prefix*
              Normally, integers are used as node names. If *prefix* is specified, this will be prepended to the integer to create the name.

−**N** *name*
              Use *name* as the name of the graph. By default, the graph is anonymous.

−**o** *outfile*
              If specified, the generated graph is written into the file *outfile*. Otherwise, the graph is written to standard out.

−**d**   Make the generated graph directed.

−**?**   Print usage information.

EXIT STATUS

gvgen exits with 0 on successful completion, and exits with 1 if given an ill-formed or incorrect flag, or if the specified output file could not be opened.

AUTHOR

Emden R. Gansner <erg@research.att.com>

SEE ALSO

gc(1), acyclic(1), gvpr(1), gvcolor(1), ccomps(1), sccmap(1), tred(1), libgraph(3)