

1. Intro. Standard input contains a list of pairs of positive integers. We set `/tmp/ez.gb` to the (undirected) graph with those edges.

```
#define maxm 10000
#include <stdio.h>
#include <stdlib.h>
#include "gb_graph.h"
#include "gb_save.h"
#include "gb_basic.h"
unsigned int u[maxm], v[maxm];
main()
{
    Graph *g;
    register int k;
    unsigned int nn;
    for (k = 0, nn = 0; k < maxm; k++) {
        if (scanf("%u%u", &u[k], &v[k]) != 2) break;
        if (u[k] > nn) nn = u[k];
        if (v[k] > nn) nn = v[k];
    }
    if (k == maxm) {
        fprintf(stderr, "Sorry, I can handle only %d edges!\n", maxm);
        exit(-1);
    }
    g = empty(nn + 1);
    for (k--; k ≥ 0; k--) gb_new_edge(g-vertices + u[k], g-vertices + v[k], 1);
    save_graph(g, "/tmp/ez.gb");
    printf("Created graph /tmp/ez.gb with %ld vertices and %ld edges.\n", g-n, g-m/2);
}
```

2. Index.

empty: 1.
exit: 1.
fprintf: 1.
g: 1.
gb_new_edge: 1.
Graph: 1.
k: 1.
main: 1.
maxm: 1.
nn: 1.
printf: 1.
save_graph: 1.
scanf: 1.
stderr: 1.
u: 1.
v: 1.
vertices: 1.

EZGRAPH

	Section	Page
Intro	1	1
Index	2	2