

1. Intro. Given a graph g with m edges, make data from which DLX2 should tell us all ways to label the vertices, using distinct labels in $\{0, 1, \dots, m\}$, so that the edges have distinct difference. (Those differences will be $\{1, \dots, m\}$).

Selected vertex labels may be prespecified on the command line, as in BACK-GRACEFUL.

```
#define encode(x) ((x) < 10 ? (x) + '0' : (x) < 36 ? (x) - 10 + 'a' : (x) < 62 ? (x) - 36 + 'A' : (x) + 99)
#define maxm 156 /* based on that encoding, but I could go higher in a pinch! */
#define maxn 100
#include <stdio.h>
#include <stdlib.h>
#include "gb_graph.h"
#include "gb_save.h"
int c;
int label; /* a label value read from argv[k] */
int prespec[maxn]; /* prespecified labels */
int verttoprespec[maxn]; /* has this vertex been prespecified? */
int prespecptr; /* how many are prespecified? */
main(int argc, char *argv[])
{
    register int i, j, k, m, n, p, x, bad;
    register Arc *a;
    register Graph *g;
    register Vertex *v, *w;
    ⟨Process the command line, and set prespec to the prespecified labelings 2⟩;
    ⟨Output the item-name line 3⟩;
    for (k = 1; k ≤ m; k++) ⟨Output the options for edge k 4⟩;
}
```

2. \langle Process the command line, and set *prespec* to the prespecified labelings 2 $\rangle \equiv$

```

if (argc < 2) {
    fprintf(stderr, "Usage: %s foo.gb [VERTEX=label...]\n", argv[0]);
    exit(-1);
}
g = restore_graph(argv[1]);
if ( $\neg$ g) {
    fprintf(stderr, "I couldn't reconstruct graph %s!\n", argv[1]);
    exit(-2);
}
m = g-m/2, n = g-n;
if (m > maxm) {
    fprintf(stderr, "Sorry, at present I require m <= %d!\n", maxm);
    exit(-3);
}
if (n > maxn) {
    fprintf(stderr, "Sorry, at present I require n <= %d!\n", maxn);
    exit(-4);
}
for (k = 2; argv[k]; k++) {
    for (i = 1; argv[k][i]; i++)
        if (argv[k][i]  $\equiv$  '=' ) break;
    if ( $\neg$ argv[k][i]  $\vee$  sscanf(&argv[k][i + 1], "%d", &label)  $\neq$  1  $\vee$  label < 0  $\vee$  label > m) {
        fprintf(stderr, "spec '%s' doesn't have the form 'VERTEX=label'!\n", argv[k]);
        exit(-3);
    }
    argv[k][i] = 0;
    for (j = 0; j < n; j++)
        if (strcmp((g-vertices + j)-name, argv[k])  $\equiv$  0) break;
    if (j  $\equiv$  n) {
        fprintf(stderr, "There's no vertex named '%s'!\n", argv[k]);
        exit(-5);
    }
    if (verttoprespec[j]) {
        fprintf(stderr, "Vertex %s was already specified!\n", (g-vertices + j)-name);
        exit(-6);
    }
    argv[k][i] = '=';
    verttoprespec[j] = 1;
    prespec[prespecptr++] = (j  $\ll$  8) + label;
}
fprintf(stderr, "OK, I've got a graph with %d vertices, %d edges, %d prespec %s.\n", n, m,
    prespecptr, prespecptr  $\equiv$  1 ? "" : "s");
printf("|");
for (k = 0; argv[k]; k++) printf("%s", argv[k]);
printf("\n");

```

This code is used in section 1.

3. There's a primary item k for each edge label, and a primary item uv for each edge. This enforces a permutation between edges and labels.

There's a secondary item $.v$ for each vertex; its color will be its label.

There's a secondary item $+k$ for each vertex label; its color will be the vertex so labeled.

⟨ Output the item-name line 3 ⟩ \equiv

```

for ( $k = 1$ ;  $k \leq m$ ;  $k++$ )  $\text{printf}(\text{"\%c\_"}, \text{encode}(k));$ 
for ( $v = g\text{-vertices}$ ;  $v < g\text{-vertices} + n$ ;  $v++$ )
  for ( $a = v\text{-arcs}$ ;  $a; a = a\text{-next}$ )
    if ( $a\text{-tip} > v$ )  $\text{printf}(\text{"\%s-\%s\_"}, v\text{-name}, a\text{-tip-name});$ 
 $\text{printf}(\text{"|"});$ 
for ( $v = g\text{-vertices}$ ;  $v < g\text{-vertices} + n$ ;  $v++$ )  $\text{printf}(\text{"\_.\%s"}, v\text{-name});$ 
for ( $k = 0$ ;  $k \leq m$ ;  $k++$ )  $\text{printf}(\text{"\_+\%c"}, \text{encode}(k));$ 
 $\text{printf}(\text{"\n"});$ 

```

This code is used in section 1.

4. `#define vrt(v) ((int)((v) - g-vertices))`

⟨ Output the options for edge k 4 ⟩ \equiv

```

{
  for ( $i = 0, j = k$ ;  $j \leq m$ ;  $i++, j++$ ) {
    for ( $v = g\text{-vertices}$ ;  $v < g\text{-vertices} + n$ ;  $v++$ )
      for ( $a = v\text{-arcs}$ ;  $a; a = a\text{-next}$ )
        if ( $a\text{-tip} > v$ ) {
          for ( $bad = p = 0$ ;  $p < \text{prespecptr}$ ;  $p++$ ) {
             $w = g\text{-vertices} + (\text{prespec}[p] \gg 8), x = \text{prespec}[p] \& \#ff;$ 
            if ( $v \equiv w$ ) {
              if ( $i \neq x$ )  $bad |= 1;$ 
              if ( $j \neq x$ )  $bad |= 2;$ 
            } else if ( $a\text{-tip} \equiv w$ ) {
              if ( $j \neq x$ )  $bad |= 1;$ 
              if ( $i \neq x$ )  $bad |= 2;$ 
            }
          }
          if ( $i \equiv x$ ) {
            if ( $v \neq w$ )  $bad |= 1;$ 
            if ( $a\text{-tip} \neq w$ )  $bad |= 2;$ 
          } else if ( $j \equiv x$ ) {
            if ( $v \neq w$ )  $bad |= 2;$ 
            if ( $a\text{-tip} \neq w$ )  $bad |= 1;$ 
          }
        }
      }
    if ( $((bad \& 1) \equiv 0)$ )  $\text{printf}(\text{"\%c\_%s-\%s\_.\%s:\%c\_.\%s:\%c\_%c:\%c\_%c:\%c\n"}, \text{encode}(k),$ 
       $v\text{-name}, a\text{-tip-name}, v\text{-name}, \text{encode}(i), a\text{-tip-name}, \text{encode}(j), \text{encode}(i),$ 
       $\text{encode}(vrt(v)), \text{encode}(j), \text{encode}(vrt(a\text{-tip}))$ );
    if ( $((bad \& 2) \equiv 0)$ )  $\text{printf}(\text{"\%c\_%s-\%s\_.\%s:\%c\_.\%s:\%c\_%c:\%c\_%c:\%c\n"}, \text{encode}(k),$ 
       $v\text{-name}, a\text{-tip-name}, v\text{-name}, \text{encode}(j), a\text{-tip-name}, \text{encode}(i), \text{encode}(j),$ 
       $\text{encode}(vrt(v)), \text{encode}(i), \text{encode}(vrt(a\text{-tip}))$ );
  }
}

```

This code is used in section 1.

5. Index.*a*: [1](#).**Arc**: [1](#).*arcs*: [3](#), [4](#).*argc*: [1](#), [2](#).*argv*: [1](#), [2](#).*bad*: [1](#), [4](#).*c*: [1](#).*encode*: [1](#), [3](#), [4](#).*exit*: [2](#).*fprintf*: [2](#).*g*: [1](#).**Graph**: [1](#).*i*: [1](#).*j*: [1](#).*k*: [1](#).*label*: [1](#), [2](#).*m*: [1](#).*main*: [1](#).*maxm*: [1](#), [2](#).*maxn*: [1](#), [2](#).*n*: [1](#).*name*: [2](#), [3](#), [4](#).*next*: [3](#), [4](#).*p*: [1](#).*prespec*: [1](#), [2](#), [4](#).*prespecptr*: [1](#), [2](#), [4](#).*printf*: [2](#), [3](#), [4](#).*restore_graph*: [2](#).*sscanf*: [2](#).*stderr*: [2](#).*strcmp*: [2](#).*tip*: [3](#), [4](#).*v*: [1](#).**Vertex**: [1](#).*vertices*: [2](#), [3](#), [4](#).*verttoprespec*: [1](#), [2](#).*vrt*: [4](#).*w*: [1](#).*x*: [1](#).

⟨ Output the item-name line 3 ⟩ Used in section 1.

⟨ Output the options for edge k 4 ⟩ Used in section 1.

⟨ Process the command line, and set *prespec* to the prespecified labelings 2 ⟩ Used in section 1.

GRACEFUL-DLX-PRESETS

| | Section | Page |
|-------------|-------------------|------|
| Intro | 1 | 1 |
| Index | 5 | 4 |