```
//  「C で学ぶデータ構造とアルゴリズム」(西原清一)オーム社, 2008
//   解図 5・B (p.214) 連結無向グラフの横型探索による走査
//   改訂版

#include  <stdio.h>
#define  N      4
#define  NON-1
#define  ROOT   999
#define  M      100
#define  OVERFLOW   -1
#define  UNDERFLOW  -2

int graph[N][N]={{0,1,1,1},{1,0,0,1},{1,0,0,1},{1,1,1,0}};
int t[N];
int q[M],f,r;     /*待ち行列*/

int enqueue(v)
{   if (f-r == 1 || f-r+M == 1) return OVERFLOW;
    q[r] = v;
    if (++r == M) r = 0;
    return 0;
}

int dequeue()
{   int v;
    if (f == r) return UNDERFLOW;
    v = q[f++];
    if (f == M) f = 0;
    return v;
}

void breadth_first()
{   int v,w;
    v = dequeue();
    if (v != -2)
    {   printf(" %d =>",v);
        for (w = 0; w < N; w++) {
            if (graph[v][w] == 1 && t[w] == NON)
                {t[w]=v; enqueue(w);}
        }
```

```
        breadth_first();
    }
}

main()
{   int i,v=0;
    for (i=0; i<N; i++) t[i] = NON;
    f = r = 0;
    enqueue(v);
    t[v] = ROOT;
    breadth_first();
    printf(" end¥n¥n");

//  for (i=0; i<N; i++) printf("t[%d]=%d, ",i,t[i]);
//  printf("end¥n¥n");
}
```