

```

/* 邻接矩阵存储的图 - BFS */

/* IsEdge(Graph, V, W) 检查<V, W>是否图Graph中的一条边, 即W是否V的邻接点。 */
/* 此函数根据图的不同类型要做不同的实现, 关键取决于对不存在的边的表示方法。*/
/* 例如对有权图, 如果不存在边被初始化为INFINITY, 则函数实现如下: */
bool IsEdge( MGraph Graph, Vertex V, Vertex W )
{
    return Graph->G[V][W]<INFINITY ? true : false;
}

/* Visited[]为全局变量, 已经初始化为false */
void BFS ( MGraph Graph, Vertex S, void (*Visit)(Vertex) )
{
    /* 以S为出发点对邻接矩阵存储的图Graph进行BFS搜索 */
    Queue Q;
    Vertex V, W;

    Q = CreateQueue( MaxSize ); /* 创建空队列, MaxSize为外部定义的常数 */
    /* 访问顶点S: 此处可根据具体访问需要改写 */
    Visit( S );
    Visited[S] = true; /* 标记S已访问 */
    AddQ(Q, S); /* S入队列 */

    while ( !IsEmpty(Q) ) {
        V = DeleteQ(Q); /* 弹出V */
        for( W=0; W<Graph->Nv; W++ ) /* 对图中的每个顶点W */
            /* 若W是V的邻接点并且未访问过 */
            if ( !Visited[W] && IsEdge(Graph, V, W) ) {
                /* 访问顶点W */
                Visit( W );
                Visited[W] = true; /* 标记W已访问 */
                AddQ(Q, W); /* W入队列 */
            }
    } /* while结束*/
}

```