ADS-Lab 3

Island Problem

Pseudo Code:

```
class Island
{
    int p[100], count;
    Island (int n)
    {
        p[n];
        for (i=0; i<n; i++)
            p[i]=i;
        count=0;
    }

    public int findparent (int u)
    {
        if (p[u] == u)
            return u;
        return p[u] = findparent (u);
    }

    public void union (int u, int y)
    {
        int root u = find parent (r)
        int root y = findparent (y)
        if (root u != root y)
        {
            p[root u] = root y;
            count --;
        }
    }

    public void setCount (int n)
    {
        count = n;
    }

    public int count()
    {
        return count;
    }
};
```

```
int  numIslands (vector <vector <int >> mat)
{
    int  count=0
    int  m = mat.size();
    int  n = mat[0].size();
    for (int i=0; i<m; i++)
        for (int j=0; j<n; j++)
            if (mat[i][j])
                count++;
    Island  is = new Island (m×n);
    is. set count (count);
    for (int i=0; i<m; i++) {
        for (int j=0; j<n; j++) {
            if (mat[i][j]! =0) {
                if (i>0 && mat[j-1][j] !=0)
                    is.union (n* (i+j), n* (i-1)+j);
                if (i<m-1 && mat[i+1][j]!=0)
                    is.union (n* i+j, n* (i+1)+j);
                if (j>0 && mat[i][j-1])
                    is.union (n* i+j, n* i+j-1);
                if (j<n-1 && mat[i][j+1])
                    is.union (n* i+j, n*i+j+1);
                if (i>0 && j>0 && mat[i-1][j-1])
                    is.union (n* i+j, n* (i-1)+j-1);
                if (i<m-1 && j<n-1 && mat[i+1][j+1])
                    is.union (n* i+j, n* (i+1)+j+1);
                if (i>0 && j<n-1 && mat[i-1][j+1])
                    is.union (n* i+j, n* (i-1)+j+1);
                if (i<m-1 && j>0 && mat[i+1][j-1])
                    is.union (n* i+j, n* (i+1)+j-1);
            }
        }
    }
    return is.count();
}
```