

### Lab 3

```
int vis[1001][1001] = {False}
```

```
int an [1001][1001];
```

```
int m, n;
```

```
int parent[1001][1001] = {-1}
```

```
bool isValid (int x, int y) {
```

```
    if (x > m || y > n || y < 0 || x <= 0)
```

```
        return false;
```

```
    if (vis[x][y] == True || an[x][y] == 0)
```

```
        return false;
```

```
    return true;
```

y



```
void union (int x, int y, int par) {
```

```
    vis[x][y] = True;
```

```
    parent[x][y] = par;
```

```
    if (isValid (x-1, y+1))
```

```
        union (x-1, y+1, par);
```

```
    if (isValid (x, y+1))
```

```
        union (x, y+1, par);
```

```
    if (isValid (x+1, y+1))
```

```
        union (x+1, y+1, par);
```

```
    if (isValid (x, y-1))
```

```
        union (x, y-1, par);
```

```
    if (isValid (x+1, y-1))
```

```
        union (x+1, y-1, par);
```

```
    if (isValid (x+1, y))
```

```
        union (x+1, y, par);
```

```
    if (isValid (x-1, y-1))
```

```
        union (x-1, y-1, par);
```

```
}
```

```
int main() {
```

```
cin >> m >> n;
```

```
for i = 1 to m;
```

```
    for j = 1 to n
```

```
        cin >> arr[i][j]
```

```
    for i = 1 to m
```

```
        for j = 1 to n
```

```
            if (arr[i][j] == 1 && !vis[i][j])
```

```
                union(i, j, i)
```

```
map<int, int> mp;
```

```
for i = 1 to m
```

```
    for j = 1 to n
```

```
        if (parent[i][j] != -1)
```

```
            mp[parent[i][j]]++;
```

```
cout << mp.size() << endl;
```

```
}
```