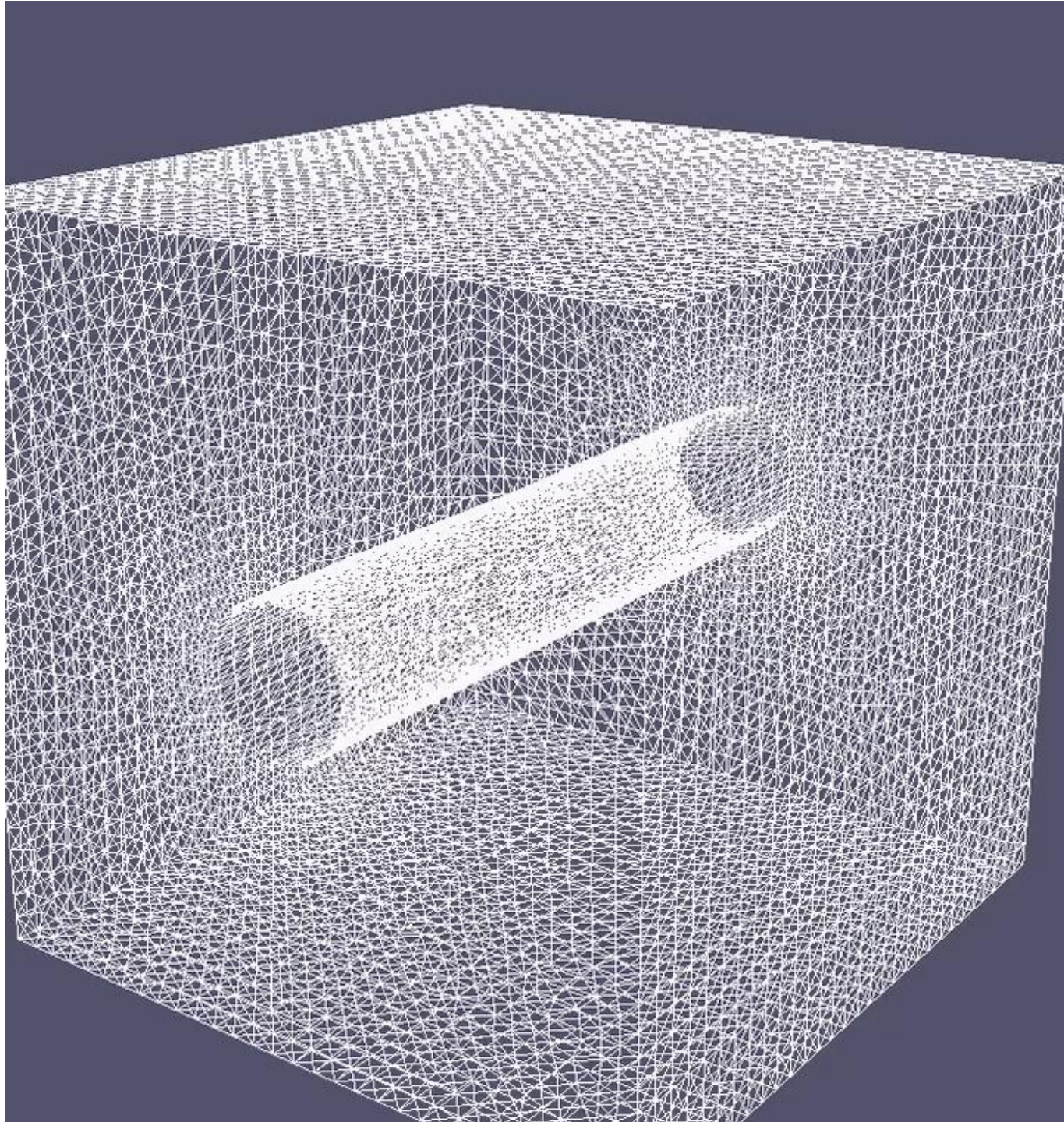


Lecture of FreeFEM++





Running Sample files

サンプルファイルの実行

- Laplace.edp
- diffusion.edp
- convection.edp
- tunnel.edp
- LapComplexEigenValue.edp
- Mesh_square.edp
- Mesh_circle.edp
- Mesh_circle_in_square.edp

Mesh generation (Mesh_square.edp)

メッシュ生成

```
border a0(t=1,0){ x=0; y=t; label=1;}
```

```
border a1(t=0,1){ x=t; y=0; label=2;}
```

```
border a2(t=0,1){ x=1; y=t; label=3;}
```

```
border a3(t=1,0){ x=t; y=1; label=4;}
```

```
int n=5;
```

```
Mesh Th
```

```
=buildmesh( a0(10*n)+a1(10*n)+a2(10*n)+a3(10*n));
```

Mesh generation (Mesh_square.edp)

メッシュ生成

```
border a0(t=1,0){ x=0; y=t; label=1;}
```

```
border a1(t=0,1){ x=t; y=0; label=2;}
```

```
border a2(t=0,1){ x=1; y=t; label=3;}
```

```
border a3(t=1,0){ x=t; y=1; label=4;}
```

```
int n=5;
```

```
Mesh Th
```

```
=buildmesh( a0(10*n)+a1(10*n)+a2(10*n)+a3(10*n));
```

Command for definitions of borders

境界を定義する際のコマンド



```
border a0(t=1,0){ x=0; y=t; label=1;}
```



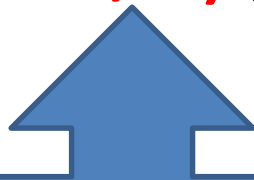
```
border a0(t=1,0){ x=0; y=t; label=1;}
```



The name of borders, and you can use any words

任意に決めることが可能な境界の名前

```
border a0(t=1,0){ x=0; y=t; label=1;}
```



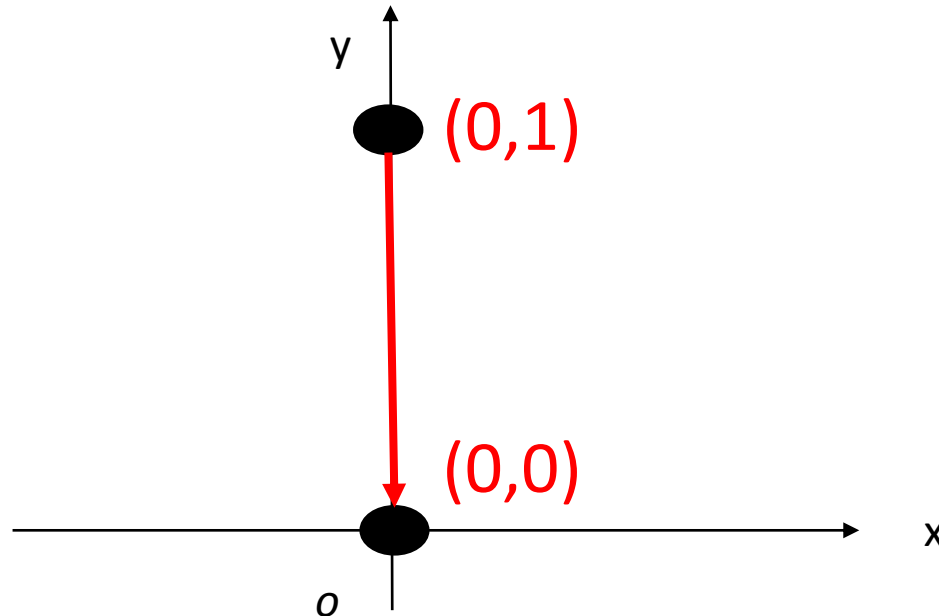
A parameter range from 1 to 0, you can use any numbers.

任意に決めることが可能な1から0に変化するパラメーター

positions of borders

境界の位置座標

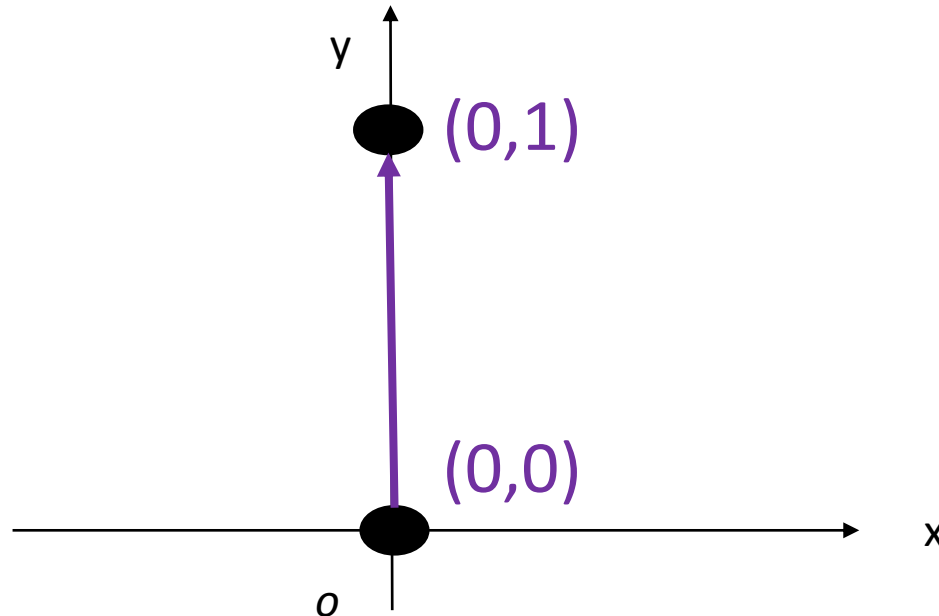
border a0(t=1,0){ **x=0; y=t; label=1;**}

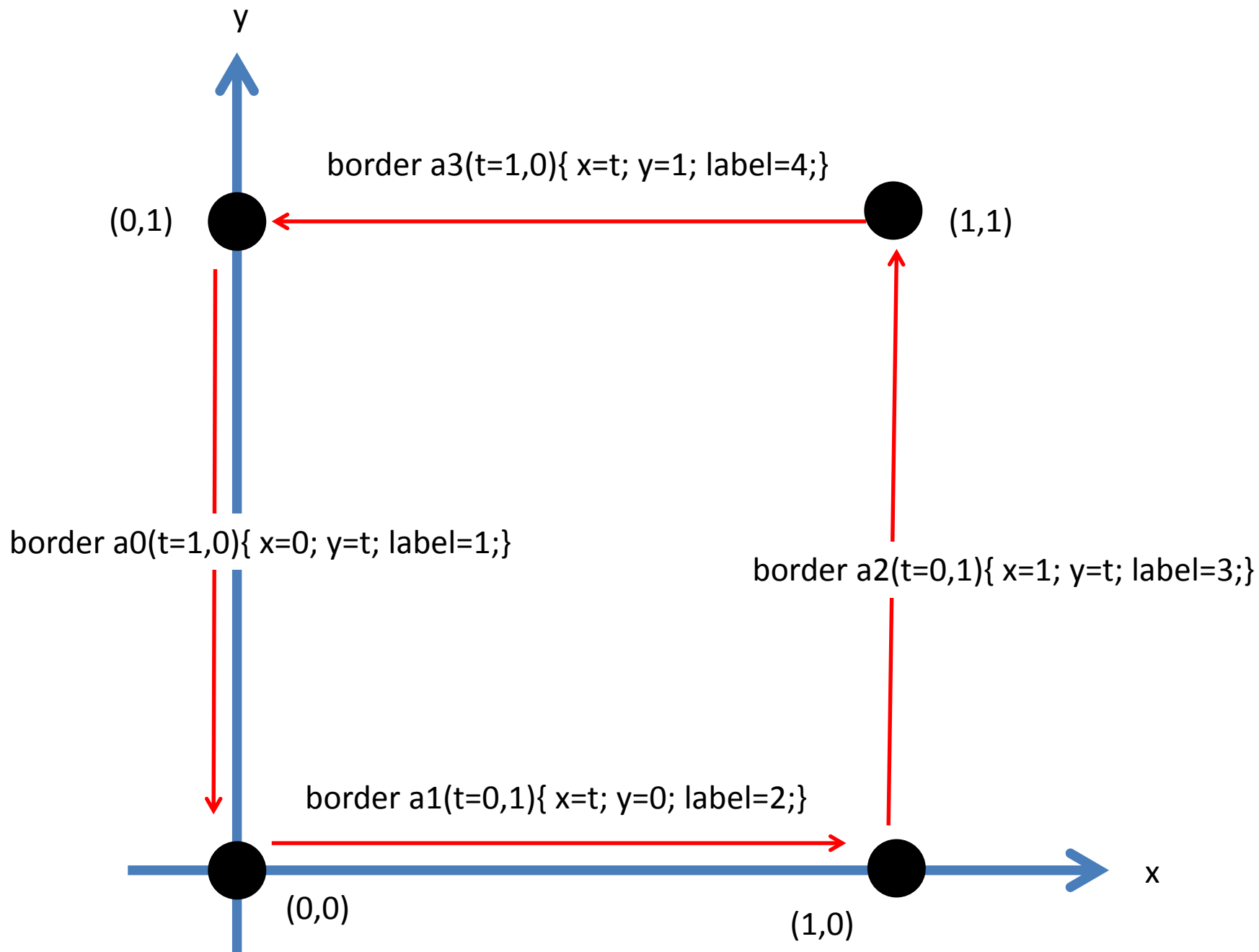


positions of borders

境界の位置座標

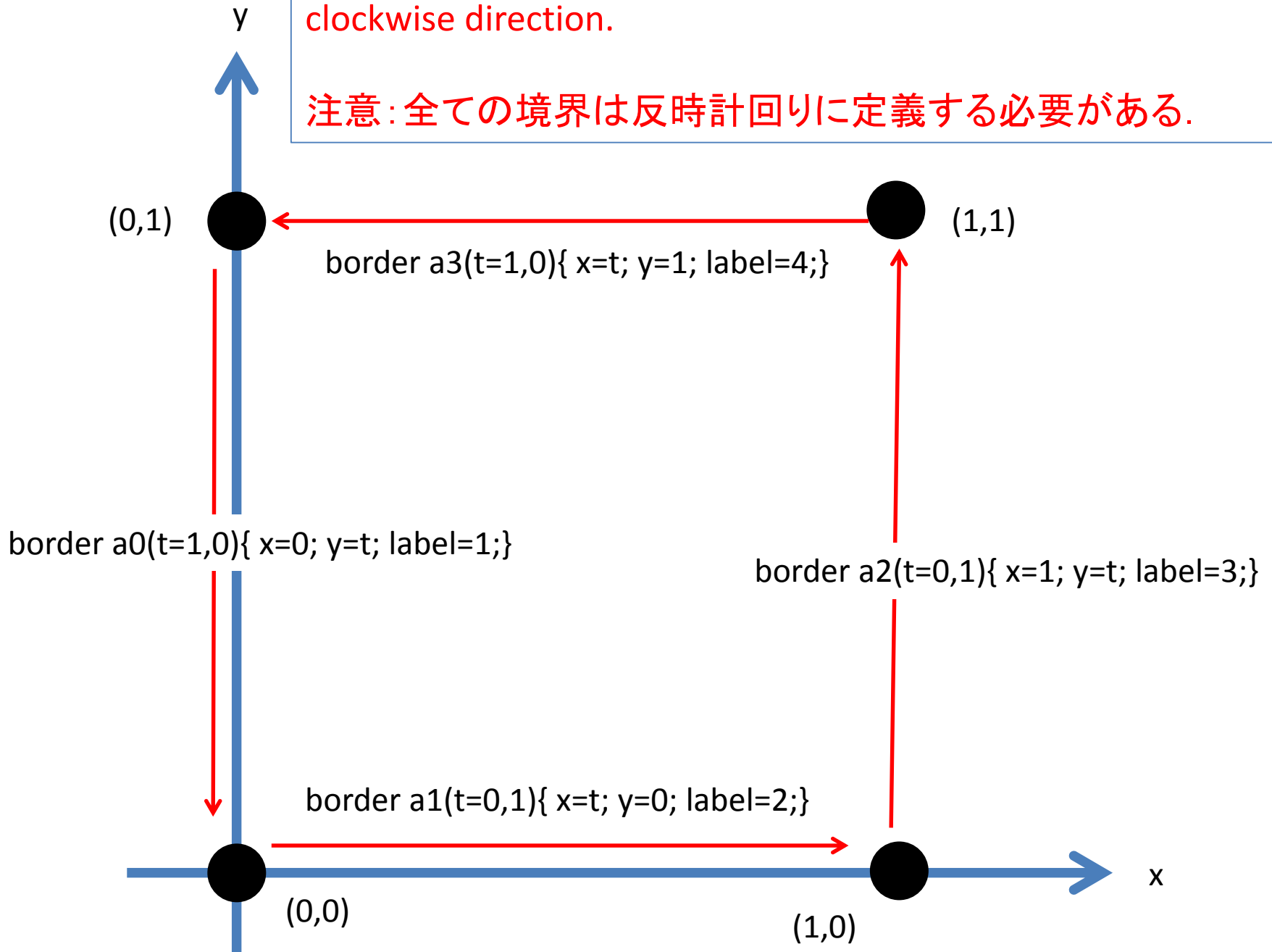
border a0(t=0,1){ x=0; y=t; label=1;}





Notations: All the borders should be defined in the anti-clockwise direction.

注意: 全ての境界は反時計回りに定義する必要がある。



Mesh generation (Mesh_square.edp)

メッシュ生成

```
border a0(t=1,0){ x=0; y=t; label=1;}
```

```
border a1(t=0,1){ x=t; y=0; label=2;}
```

```
border a2(t=0,1){ x=1; y=t; label=3;}
```

```
border a3(t=1,0){ x=t; y=1; label=4;}
```

```
int n=5;
```

```
Mesh Th
```

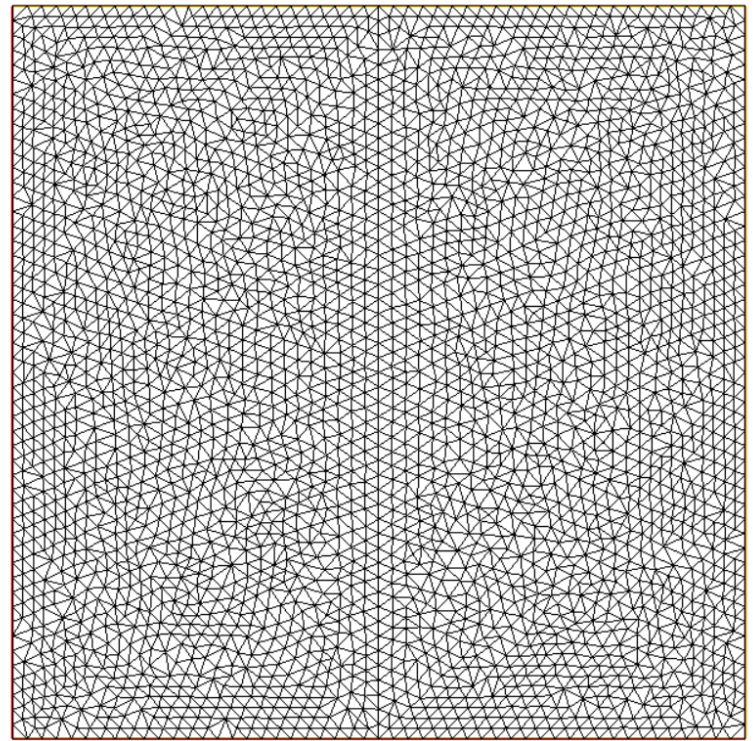
```
=buildmesh( a0(10*n)+a1(10*n)+a2(10*n)+a3(10*n));
```

Command for definitions of meshes

メッシュを定義する際のコマンド

Mesh Th

=**buildmesh**(a0(10*n)+a1(10*n)+a2(10*n)+a3(10*n));



The name of meshes.
You can use any words.

任意に決めることが可能な
メッシュの名前



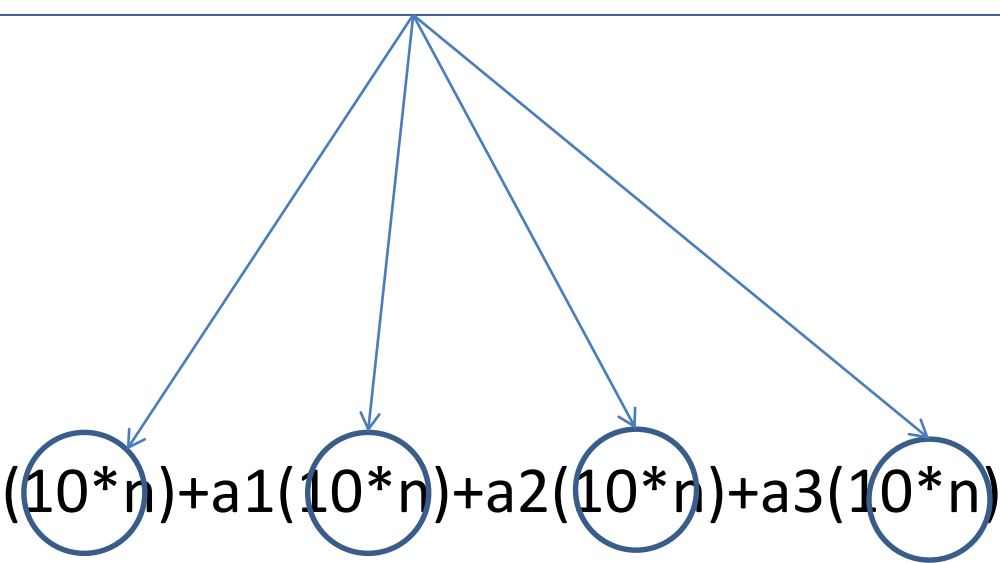
Mesh Th

=buildmesh(a0(10*n)+a1(10*n)+a2(10*n)+a3(10*n));

**Fineness of meshes should be defined by integer numbers.
メッシュの細かさ・粗さは整数で定義する必要がある。**

Mesh Th

=buildmesh(a0(10*n)+a1(10*n)+a2(10*n)+a3(10*n));



```
graph TD; A[Fineness of meshes should be defined by integer numbers.  
メッシュの細かさ・粗さは整数で定義する必要がある。] --> B((10*n)); A --> C((10*n)); A --> D((10*n)); A --> E((10*n));
```

Mesh generation (Mesh_square.edp)

メッシュ生成

```
border a0(t=1,0){ x=0; y=t; label=1;}  
border a1(t=0,1){ x=t; y=0; label=2;}  
border a2(t=0,1){ x=1; y=t; label=3;}  
border a3(t=1,0){ x=t; y=1; label=4;}
```

Definitions of boundaries
境界の定義

```
int n=5;  
Mesh Th  
=buildmesh( a0(10*n)+a1(10*n)+a2(10*n)+a3(10*n));
```

Mesh Generating
メッシュ生成

To define inside boundary (Mesh_circle_in_square.edp)
領域内に境界がある場合

```
border a0(t=1,0){ x=0; y=t; label=1;}
```

```
border a1(t=0,1){ x=t; y=0; label=2;}
```

```
border a2(t=0,1){ x=1; y=t; label=3;}
```

```
border a3(t=1,0){ x=t; y=1; label=4;}
```

```
border a4(t=0,-2*pi){ x=0.5+0.1*cos(t); y=0.5+0.1*sin(t); label=1;}
```

```
int n=5;
```

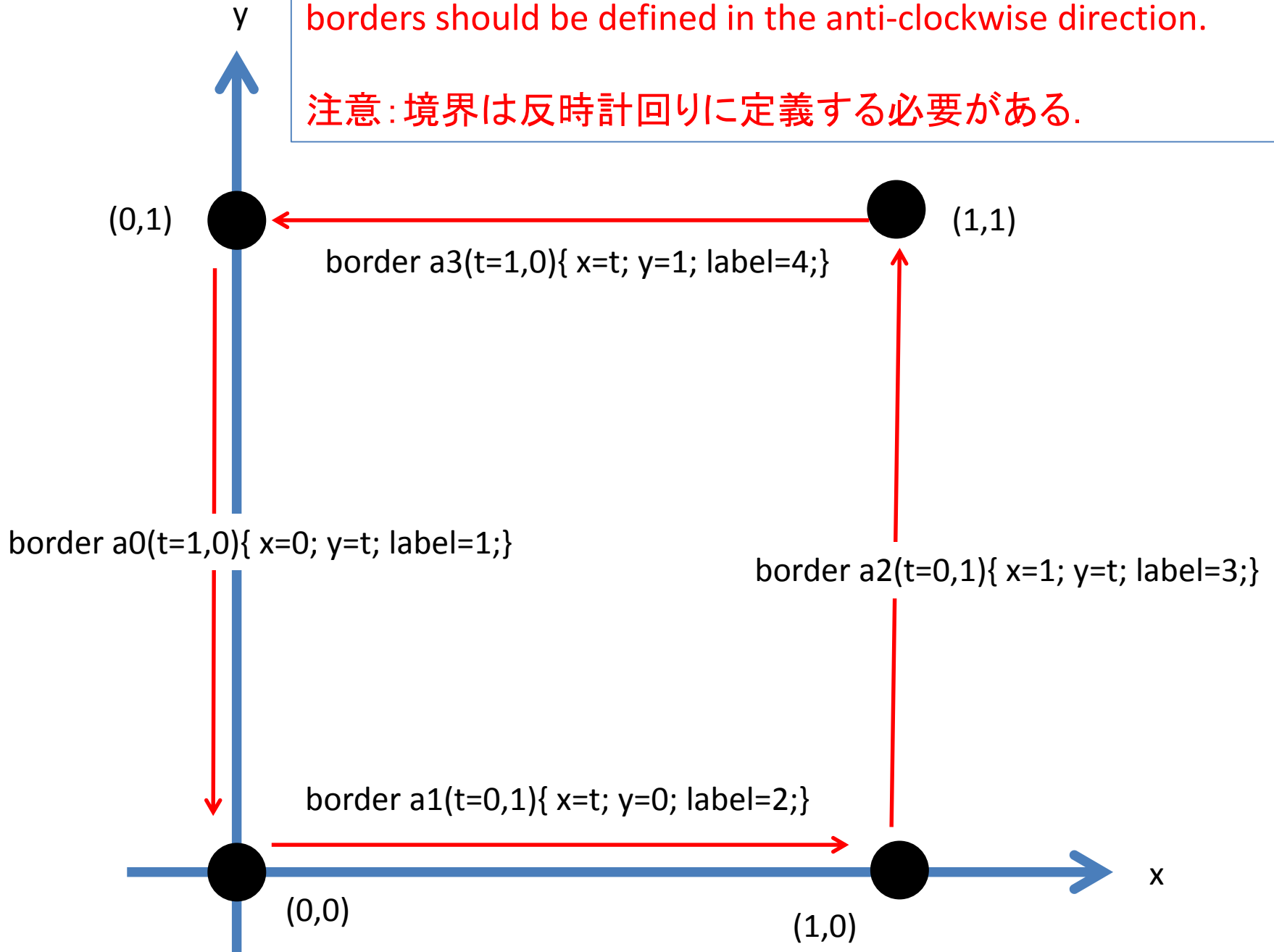
```
mesh Th=
```

```
buildmesh( a0(10*n)+a1(10*n)+a2(10*n)+a3(10*n)+a4(10*n));
```

Notations:

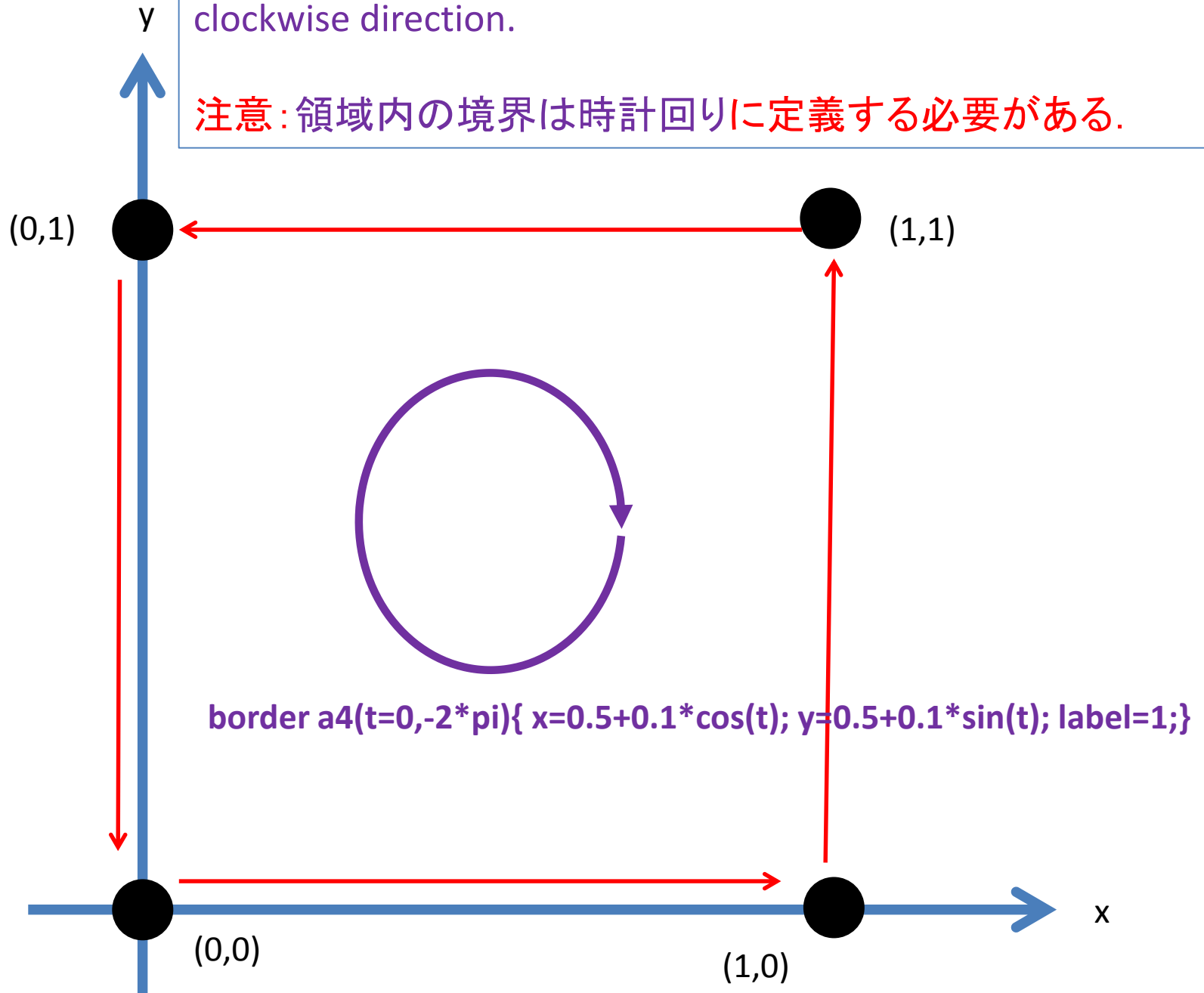
borders should be defined in the anti-clockwise direction.

注意: 境界は反時計回りに定義する必要がある.

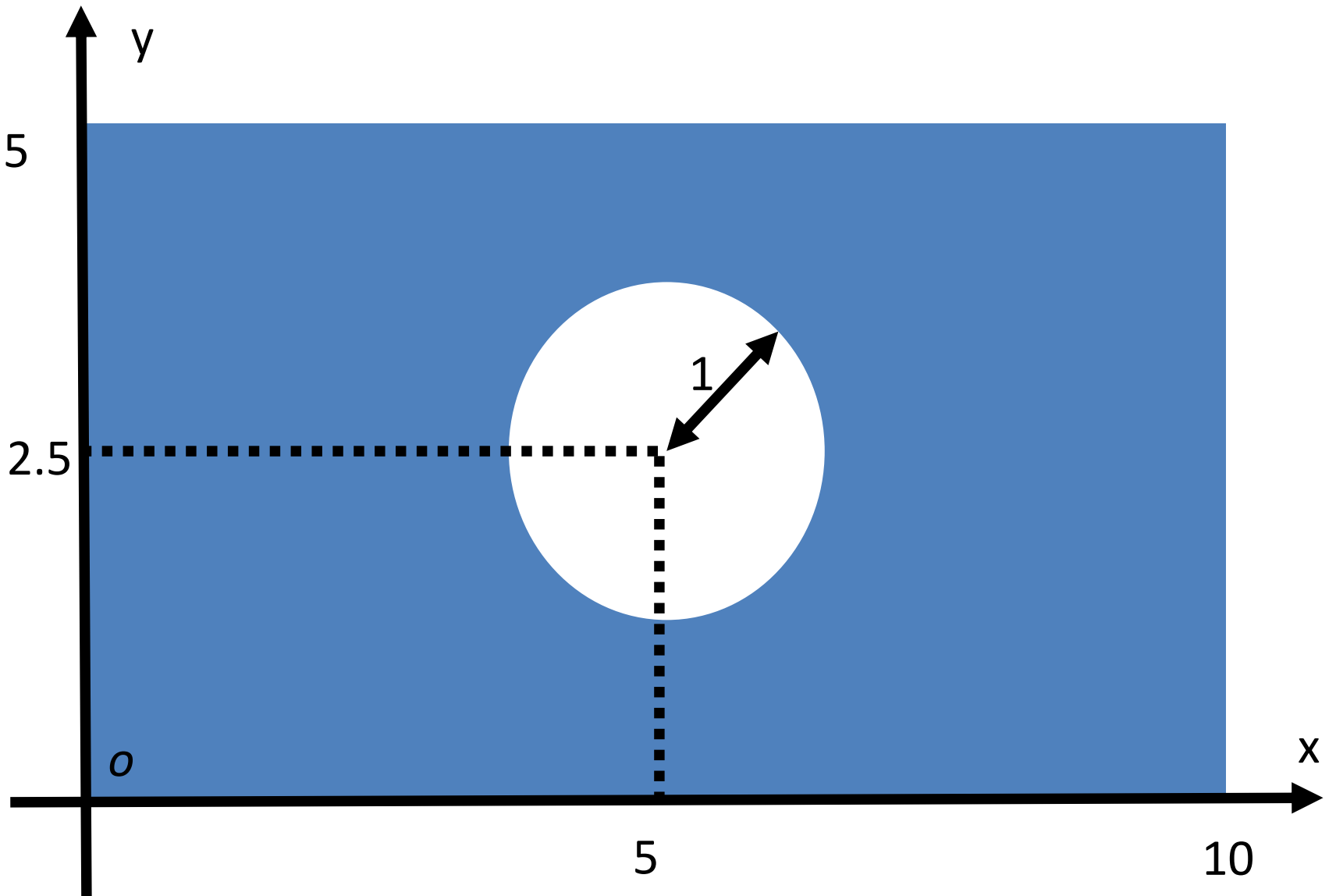


Notations: definitions of **inside** borders should be defined in the clockwise direction.

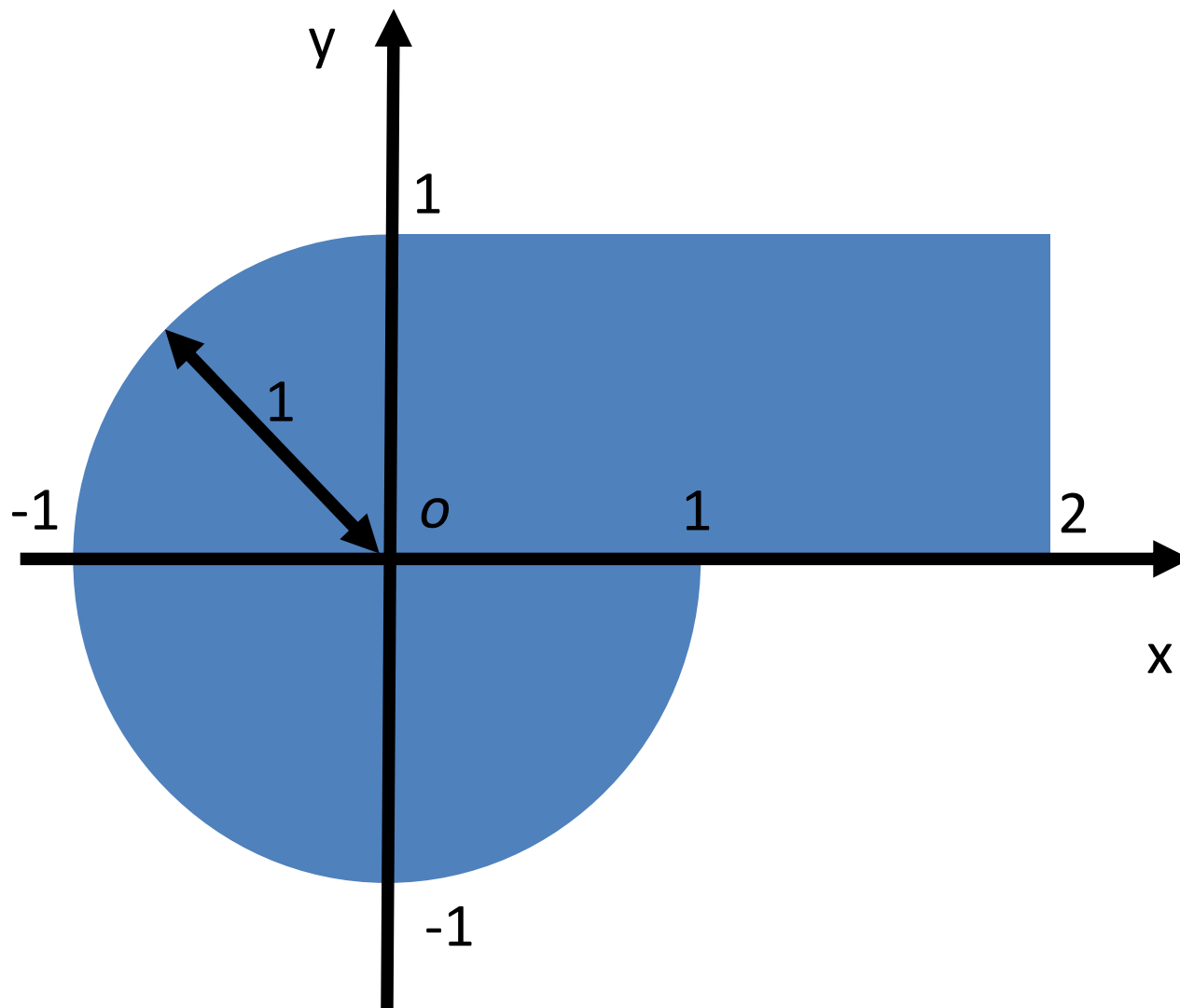
注意: 領域内の境界は時計回りに定義する必要がある.



Problem1



Problem2



Special Problem

1. Plot the mesh of “aerofoil.edp”
“aerofoil.edp”のメッシュを生成せよ
2. Change Angle of Attack using ??
回転行列を使って迎え角を変化させよ

