

```
In [1]: import numpy as np
```

```
In [2]: # standard BCC lattice:  
a=np.array([[1,1,1],[-1,-1,1],[1,-1,-1]])/2  
a/=np.sqrt(np.dot(a[0],a[0]))
```

```
In [3]: # compute all scalar products:  
for i in range(len(a)):  
    for j in range(len(a)):  
        print(np.dot(a[i],a[j]))
```

```
1.0000000000000002  
-0.3333333333333334  
-0.3333333333333334  
-0.3333333333333334  
1.0000000000000002  
-0.3333333333333334  
-0.3333333333333334  
-0.3333333333333334  
1.0000000000000002
```

```
In [4]: # AMBER lattice  
b=np.zeros((3,3))  
b[0,0]=1  
b[1,0]=-1/3  
b[1,1]=np.sqrt(8/9)  
b[2,0]=-1/3  
b[2,1]=-np.sqrt(2/9)  
b[2,2]=np.sqrt(2/3)  
b
```

```
Out [4]: array([[ 1.          ,  0.          ,  0.          ],  
               [-0.33333333,  0.94280904,  0.          ],  
               [-0.33333333, -0.47140452,  0.81649658]])
```

```
In [5]: # compute all scalar products:  
for i in range(len(b)):  
    for j in range(len(b)):  
        print(np.dot(b[i],b[j]))
```

```
1.0  
-0.3333333333333333
```

-0.3333333333333333  
-0.3333333333333333  
1.0  
-0.3333333333333333  
-0.3333333333333333  
-0.3333333333333333  
1.0