1 Define operators

```
(%i1) kill(all);
(%o0) done
```

```
(%i1) batchload("F:/Paper308/orbitals2.wxm")$
0  N(R): 1
1  N(R): 1
2  N(R): 1
3  N(R): 1
4  N(R): 1
5  N(R): 1
0 l=0, ml=0,  N(Y): 1,    <Y[3],Y[0]>: 0
1 l=1, ml=-1,  N(Y): 1,    <Y[3],Y[1]>: 0
2 l=1, ml=0,  N(Y): 1,    <Y[3],Y[2]>: 0
3 l=1, ml=1,  N(Y): 1,    <Y[3],Y[3]>: 1
4 l=2, ml=-2,  N(Y): 1,    <Y[3],Y[4]>: 0
5 l=2, ml=-1,  N(Y): 1,    <Y[3],Y[5]>: 0
6 l=2, ml=0,  N(Y): 1,    <Y[3],Y[6]>: 0
7 l=2, ml=1,  N(Y): 1,    <Y[3],Y[7]>: 0
8 l=2, ml=2,  N(Y): 1,    <Y[3],Y[8]>: 0
9 l=3, ml=-3,  N(Y): 1,    <Y[3],Y[9]>: 0
10 l=3, ml=-2,  N(Y): 1,    <Y[3],Y[10]>: 0
11 l=3, ml=-1,  N(Y): 1,    <Y[3],Y[11]>: 0
12 l=3, ml=0,  N(Y): 1,    <Y[3],Y[12]>: 0
13 l=3, ml=1,  N(Y): 1,    <Y[3],Y[13]>: 0
14 l=3, ml=2,  N(Y): 1,    <Y[3],Y[14]>: 0
15 l=3, ml=3,  N(Y): 1,    <Y[3],Y[15]>: 0
```

2 Transition matrix elements mu_Z

```
(%i2) i: -1$
      for l1: 0 thru 2 do (
      for m1: -l1 thru l1 do (
         i: i+1,
         j: -1,
         for l2: 0 thru 2 do (
         for m2: -l2 thru l2 do (
            j: j+1,
            op: cos(theta),
      if l2>l1 then (
            me: radcan(Ex22(Y[i], op, Y[j])),
            if me # 0 then (
               printf (true, "l,m: ~2d,~2d --> ~2d,~2d :~%", l1, m1, l2,
               print(me)
      ))))));
```

```
l,m:  0, 0 -->  1, 0 :
 1
---
√3
l,m:  1,-1 -->  2,-1 :
 1
---
√5
l,m:  1, 0 -->  2, 0 :
  2
-----
√3 √5
l,m:  1, 1 -->  2, 1 :
 1
---
√5
(%o3) done
```

3 Transition matrix elements mu_X

```
(%i4) i: -1$
      for l1: 0 thru 2 do (
      for m1: -l1 thru l1 do (
         i: i+1,
         j: -1,
         for l2: 0 thru 2 do (
         for m2: -l2 thru l2 do (
            j: j+1,
            op: sin(theta)*cos(phi),
      if l2>l1 then (
            me: radcan(Ex22(Y[i], op, Y[j])),
            if me # 0 then (
               printf (true, "l,m: ~2d,~2d --> ~2d,~2d :~%", l1, m1, l2,
               print(me)
      ))))));
l,m:  0, 0 -->  1,-1 :
    1
- ------
  sqrt(2) sqrt(3)
l,m:  0, 0 -->  1, 1 :
    1
- ------
  sqrt(2) sqrt(3)
l,m:  1,-1 -->  2,-2 :
  1
- -----
  sqrt(5)
l,m:  1,-1 -->  2, 0 :
    1
------------
sqrt(2) sqrt(3) sqrt(5)
l,m:  1, 0 -->  2,-1 :
    1
- ------
  sqrt(2) sqrt(5)
l,m:  1, 0 -->  2, 1 :
    1
- ------
  sqrt(2) sqrt(5)
l,m:  1, 1 -->  2, 0 :
    1
------------
sqrt(2) sqrt(3) sqrt(5)
l,m:  1, 1 -->  2, 2 :
  1
- -----
  sqrt(5)
 (%o5) done
```

4 Transition matrix elements mu_Y

```
(%i6) i: -1$
      for l1: 0 thru 2 do (
      for m1: -l1 thru l1 do (
         i: i+1,
         j: -1,
         for l2: 0 thru 2 do (
         for m2: -l2 thru l2 do (
            j: j+1,
            op: sin(theta)*sin(phi),
      if l2>l1 then (
            me: radcan(Ex22(Y[i], op, Y[j])),
            if me # 0 then (
               printf (true, "l,m: ~2d,~2d --> ~2d,~2d :~%", l1, m1, l2,
               print(me)
      ))))));
```

```
l,m:  0, 0 -->  1,-1 :
  %i
-------
sqrt(2) sqrt(3)
l,m:  0, 0 -->  1, 1 :
   %i
- -------
  sqrt(2) sqrt(3)
l,m:  1,-1 -->  2,-2 :
%i
-------
sqrt(5)
l,m:  1,-1 -->  2, 0 :
   %i
-----------------
sqrt(2) sqrt(3) sqrt(5)
l,m:  1, 0 -->  2,-1 :
  %i
-------
sqrt(2) sqrt(5)
l,m:  1, 0 -->  2, 1 :
   %i
- -------
  sqrt(2) sqrt(5)
l,m:  1, 1 -->  2, 0 :
     %i
- -----------------
  sqrt(2) sqrt(3) sqrt(5)
l,m:  1, 1 -->  2, 2 :
  %i
- -------
  sqrt(5)
(%o7) done
```