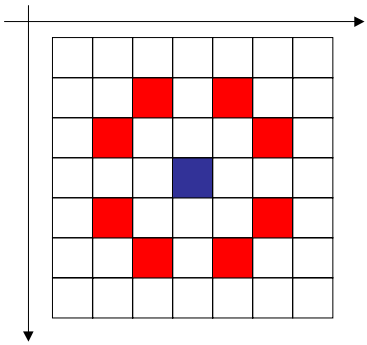
Sommario

- Definizione di ricorsione e strategie *divide et impera*
- Semplici algoritmi ricorsivi
- Merge Sort
- Quicksort
- Esempi più complessi di algoritmi ricorsivi

Il tour del cavaliere

Si desidera trovare una sequenza di mosse del cavallo tale per cui questo tocca una ed una sola volta ciascuna casella di una scacchiera N×N.

Si ricorda che un cavallo in posizione (i,j) può muovere in 8 possibili caselle:



Inizializzazioni

```
#define DIM 6
int  a[8],b[8],scacc[DIM][DIM];
void main(void)
{ int     i,   j,   result;
   a[0]=2; b[0]=1; a[1]=1; b[1]=2;
   a[2]=-1; b[2]=2; a[3]=-2; b[3]=1;
   a[4]=-2; b[4]=-1; a[5]=-1; b[5]=-2;
   a[6]=1; b[6]=-2; a[7]=2; b[7]=-1;

   for( i=0; i<DIM; i++)
      for( j=0; j<DIM; j++)
         scacc[i][j] = 0;
```


Programma principale

```
   scacc[0][0] = 1;
   result = muovi( 2, 0, 0);
   if( result == 1)
   {  for( i=0; i<DIM; i++)
      {  for( j=0; j<DIM; j++)
            printf( "%2d ",scacc[i][j]);
         printf( "\n");
      }
   } else
   {  printf( "Soluzione non trovata\n");
   }
}
```

Muovi (1)

```
int muovi( int mossa, int posx, int posy)
{  int    i,ret,newposx,newposy;
   if( mossa == (DIM*DIM+1))
      return(1);
   for( i=0; i<8; i++)
   {  newposx = posx + a[i];
      newposy = posy + b[i];
```

5 a.a. 2001/2002

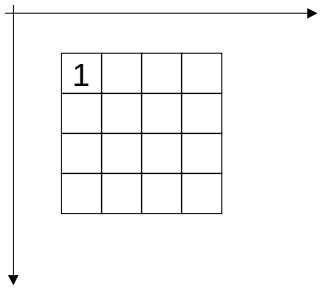
Muovi (2)

```
  if( (newposx<DIM) && (newposx>=0) &&
          (newposy<DIM) && (newposy>=0))
  {  if( scacc[newposx][newposy] == 0)
    {  scacc[newposx][newposy]=mossa;
       ret=muovi(mossa+1,newposx,newposy);
       if( ret == 0)
          scacc[newposx][newposy]=0;
       else
          return(1);
    }
  }
 }
 return(0);
}
```

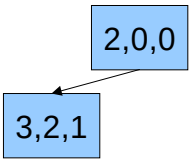
6 a.a. 2001/2002

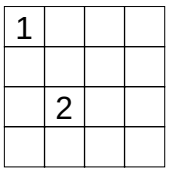
Analisi

Si supponga che N=4.



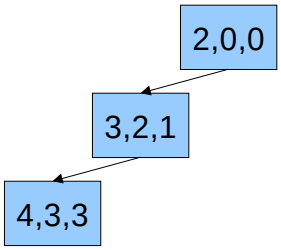
Mossa 2





Mossa 3

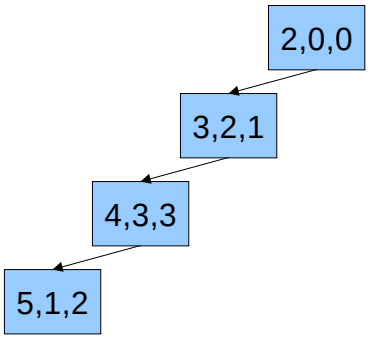
1			
	2		
			3



9 a.a. 2001/2002

Mossa 4

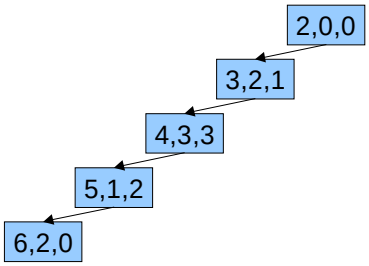
1			
		4	
	2		
			3



10 a.a. 2001/2002

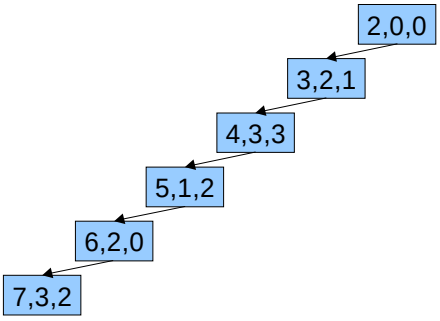
Mossa 5

1			
		4	
5	2		
			3

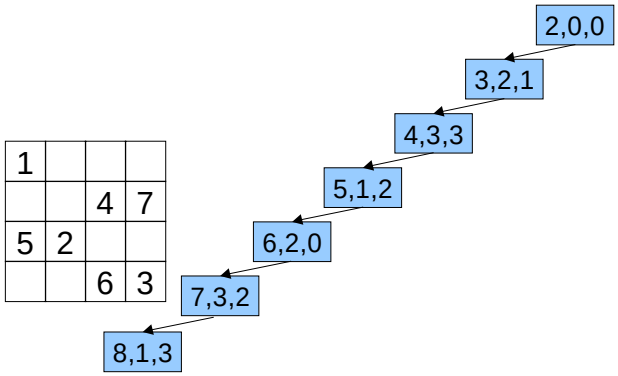


Mossa 6

1			
		4	
5	2		
		6	3



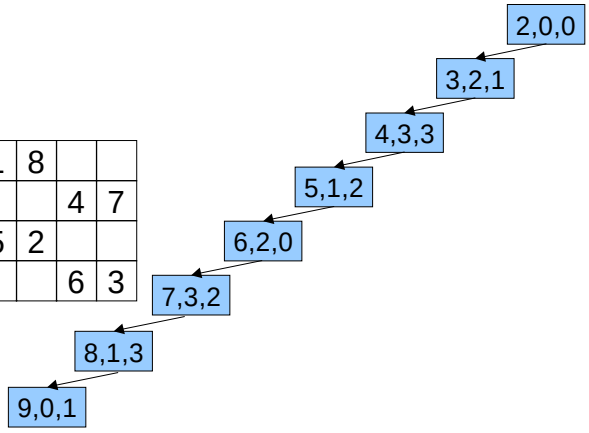
Mossa 7



1			
		4	7
5	2		
		6	3

13 a.a. 2001/2002

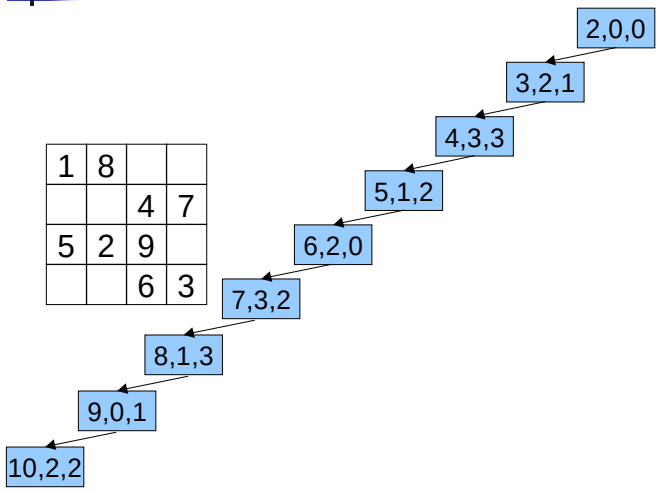
Mossa 8



1	8		
		4	7
5	2		
		6	3

14 a.a. 2001/2002

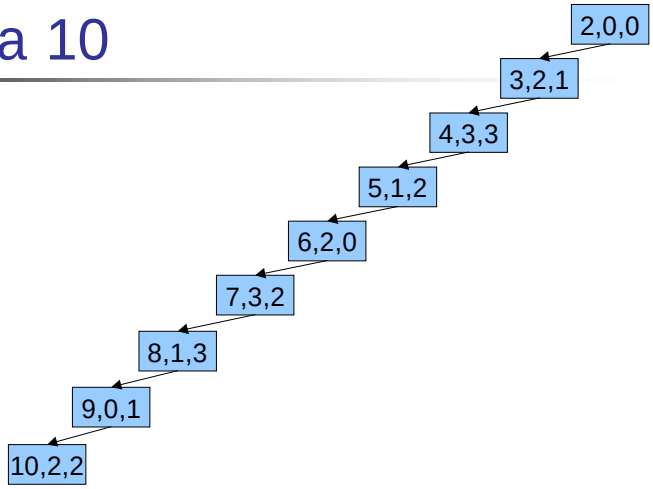
Mossa 9



1	8		
		4	7
5	2	9	
		6	3

15 a.a. 2001/2002

Mossa 10

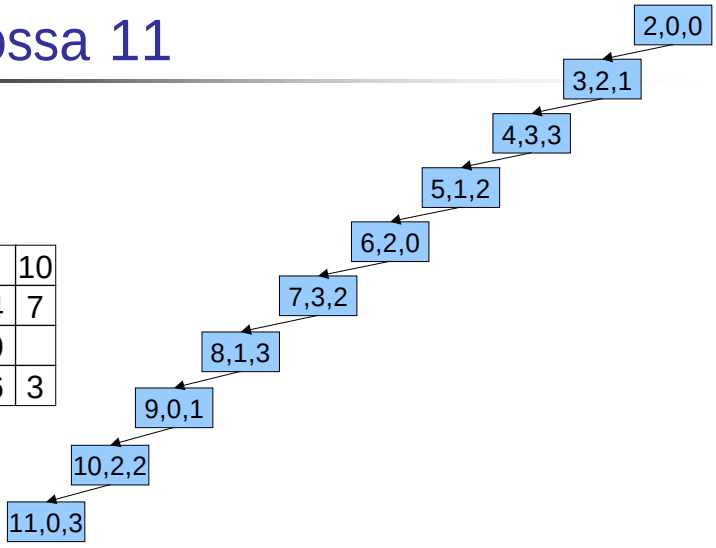


1	8		
		4	7
5	2	9	
		6	3

16 a.a. 2001/2002

Mossa 11

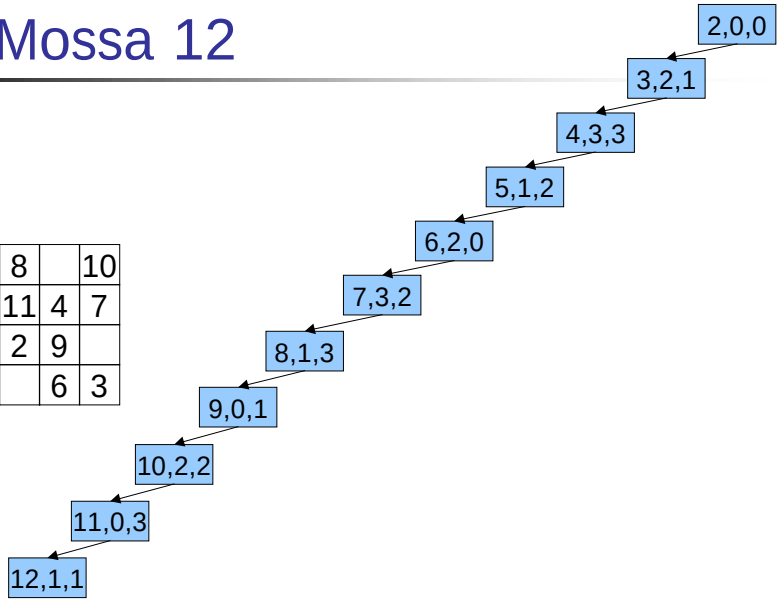
1	8		10
		4	7
5	2	9	
		6	3



17 a.a. 2001/2002

Mossa 12

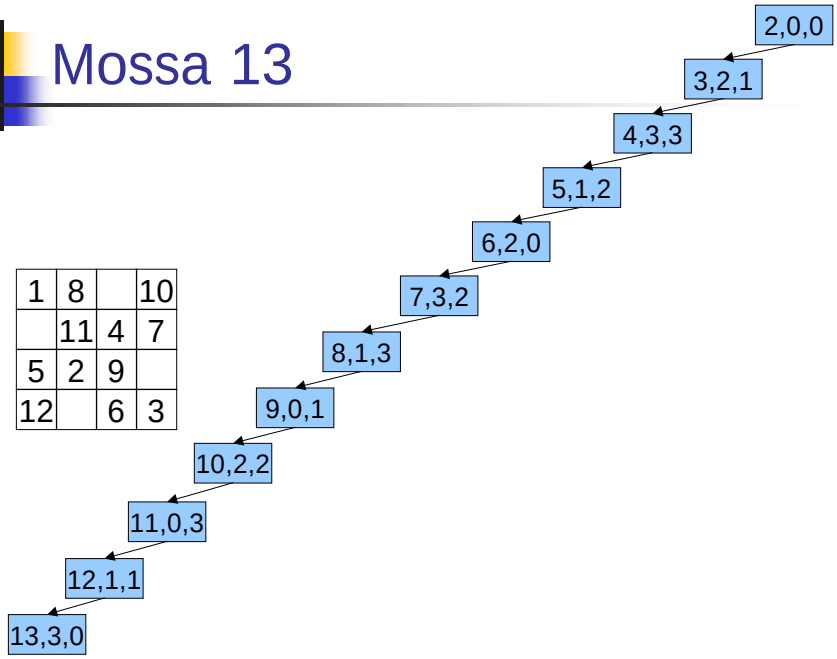
1	8		10
	11	4	7
5	2	9	
		6	3



18 a.a. 2001/2002

Mossa 13

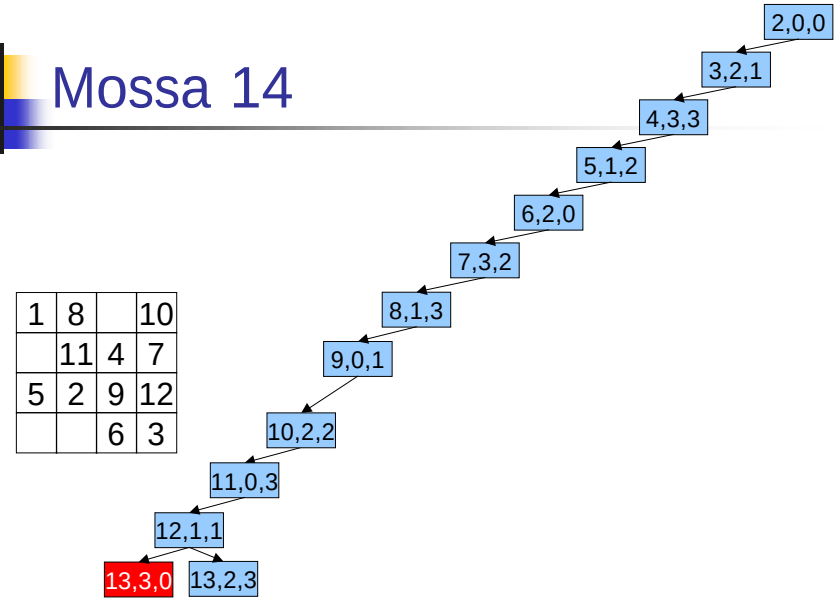
1	8		10
	11	4	7
5	2	9	
12		6	3



19 a.a. 2001/2002

Mossa 14

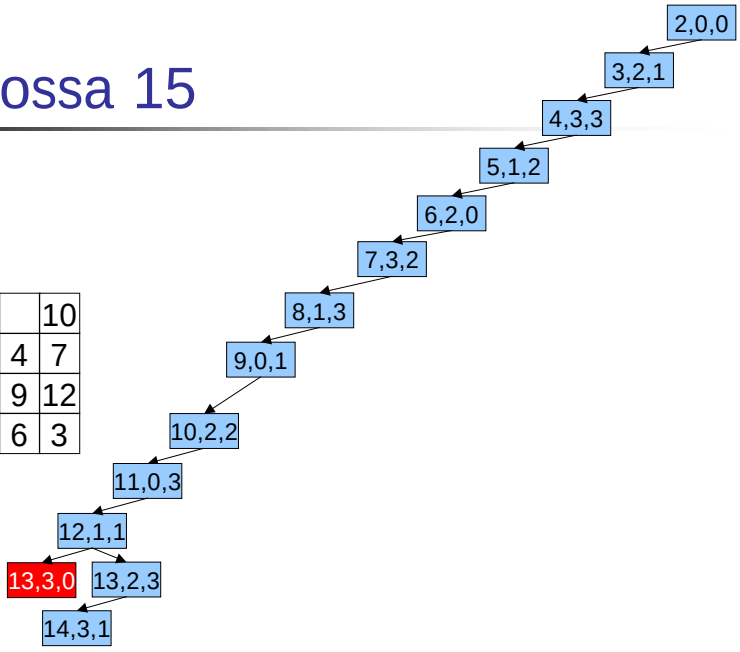
1	8		10
	11	4	7
5	2	9	12
		6	3



20 a.a. 2001/2002

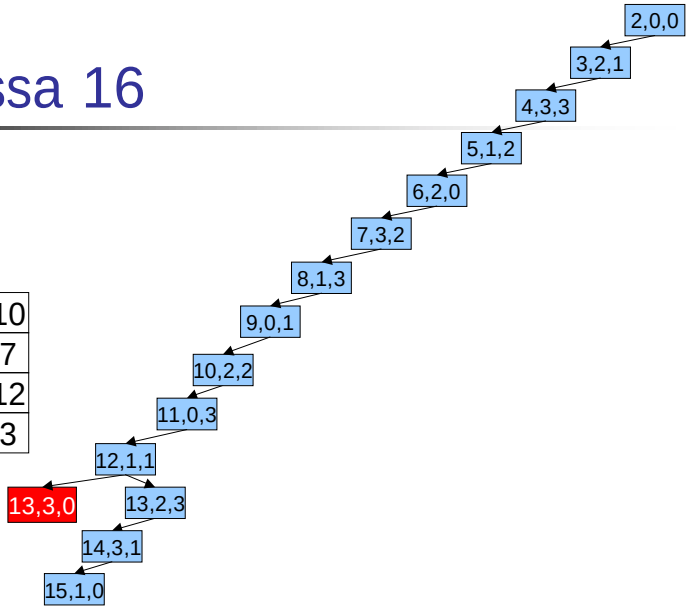
Mossa 15

1	8		10
	11	4	7
5	2	9	12
	13	6	3



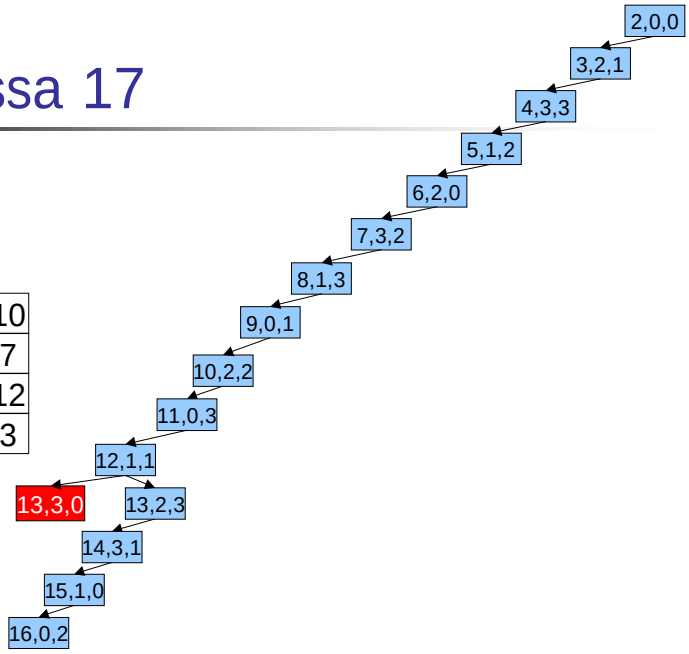
Mossa 16

1	8		10
14	11	4	7
5	2	9	12
	13	6	3



Mossa 17

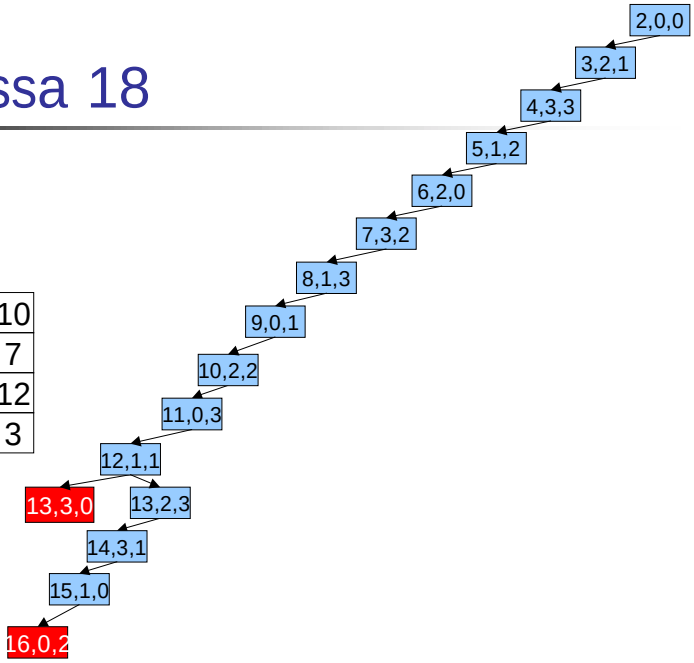
1	8	15	10
14	11	4	7
5	2	9	12
	13	6	3



23 a.a. 2001/2002

Mossa 18

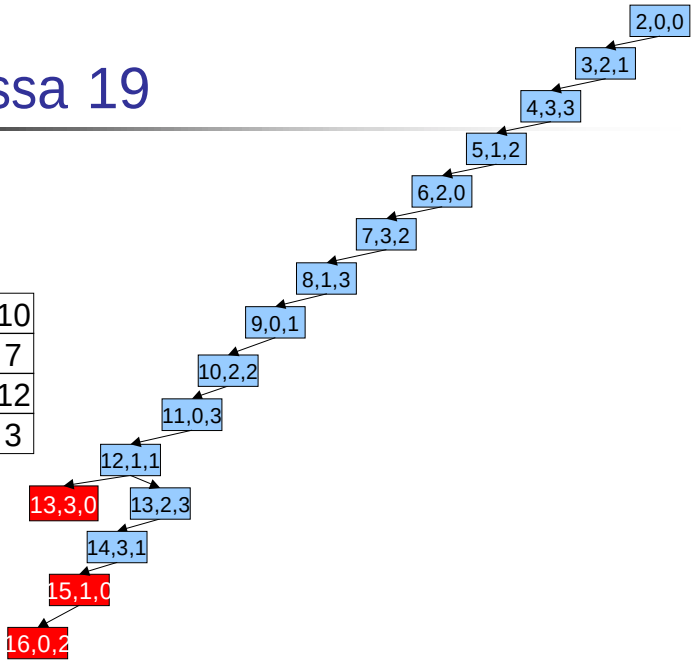
1	8		10
14	11	4	7
5	2	9	12
	13	6	3



24 a.a. 2001/2002

Mossa 19

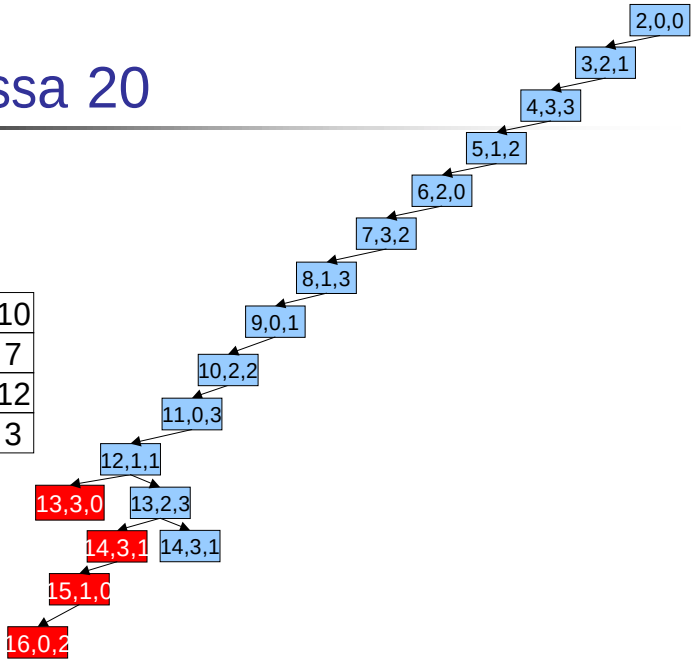
1	8		10
	11	4	7
5	2	9	12
	13	6	3



25 a.a. 2001/2002

Mossa 20

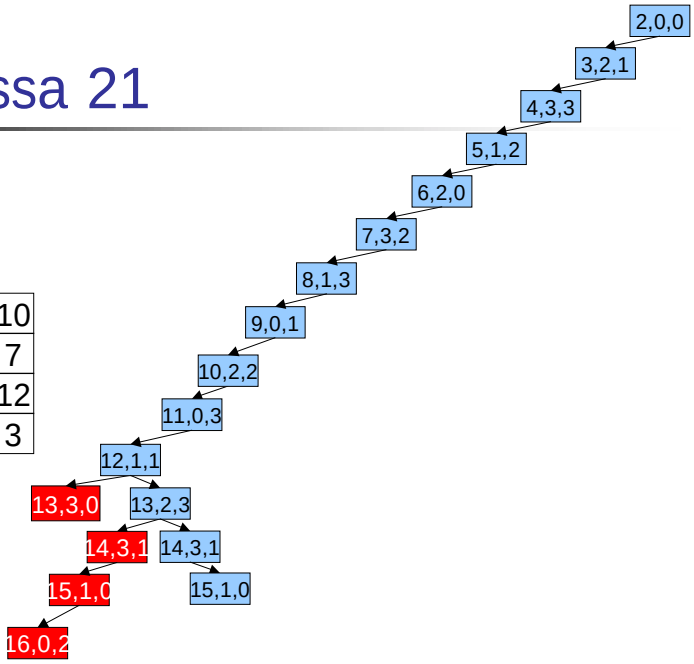
1	8	13	10
	11	4	7
5	2	9	12
		6	3



26 a.a. 2001/2002

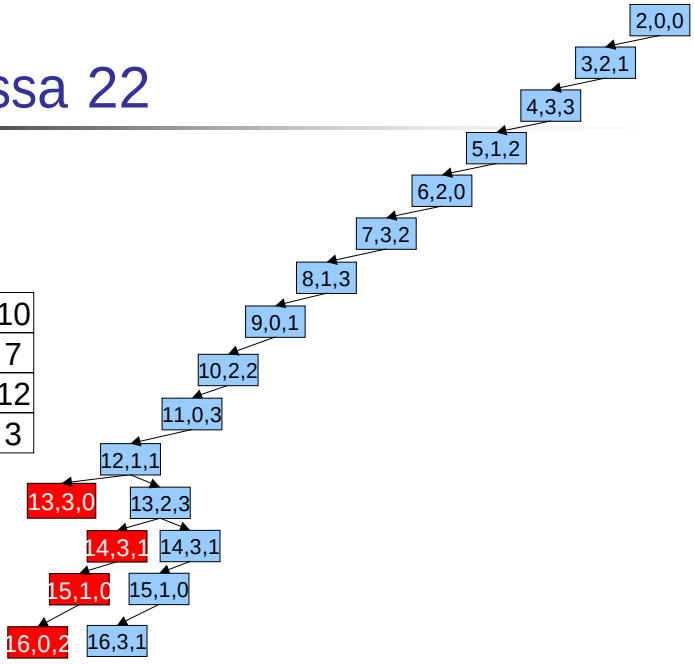
Mossa 21

1	8	13	10
14	11	4	7
5	2	9	12
		6	3



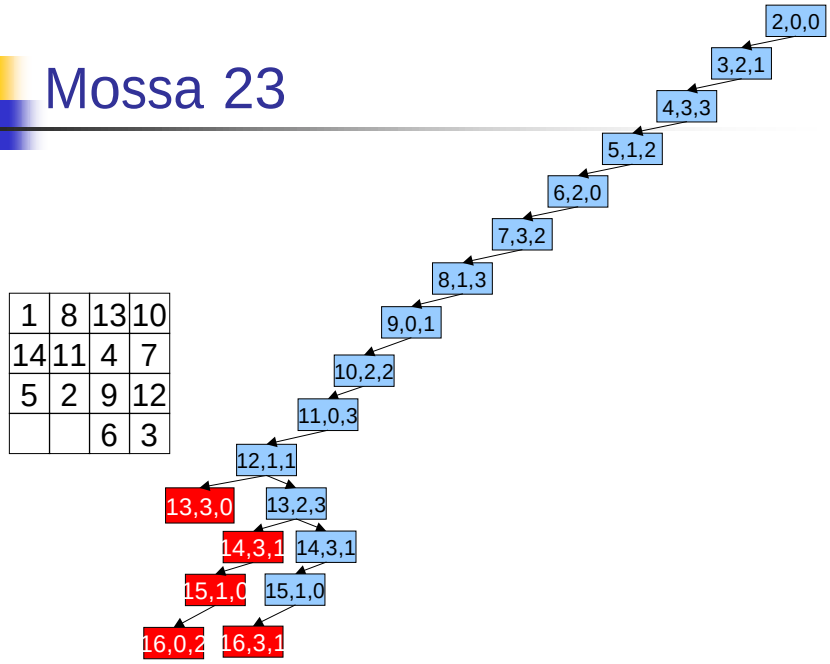
Mossa 22

1	8	13	10
14	11	4	7
5	2	9	12
	15	6	3



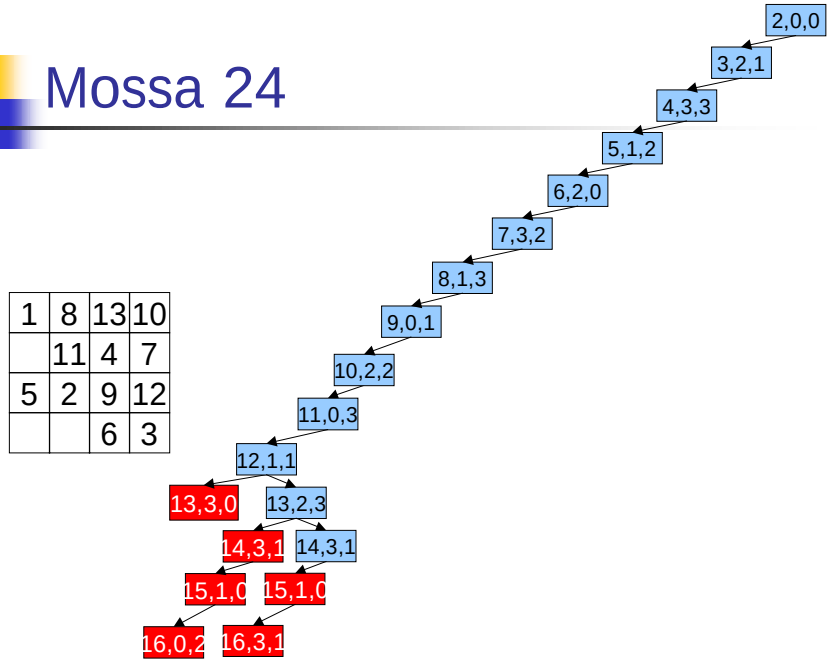
Mossa 23

1	8	13	10
14	11	4	7
5	2	9	12
		6	3



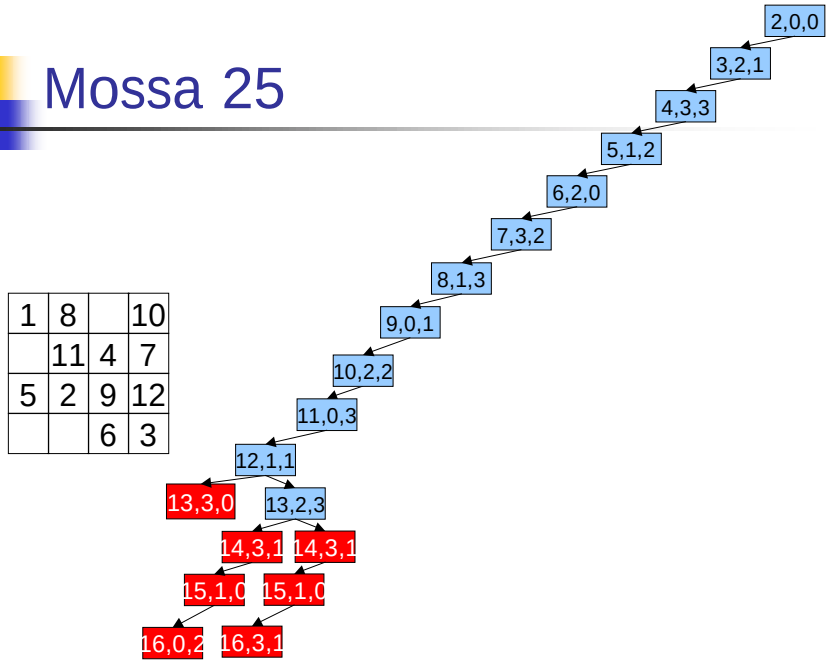
Mossa 24

1	8	13	10
	11	4	7
5	2	9	12
		6	3



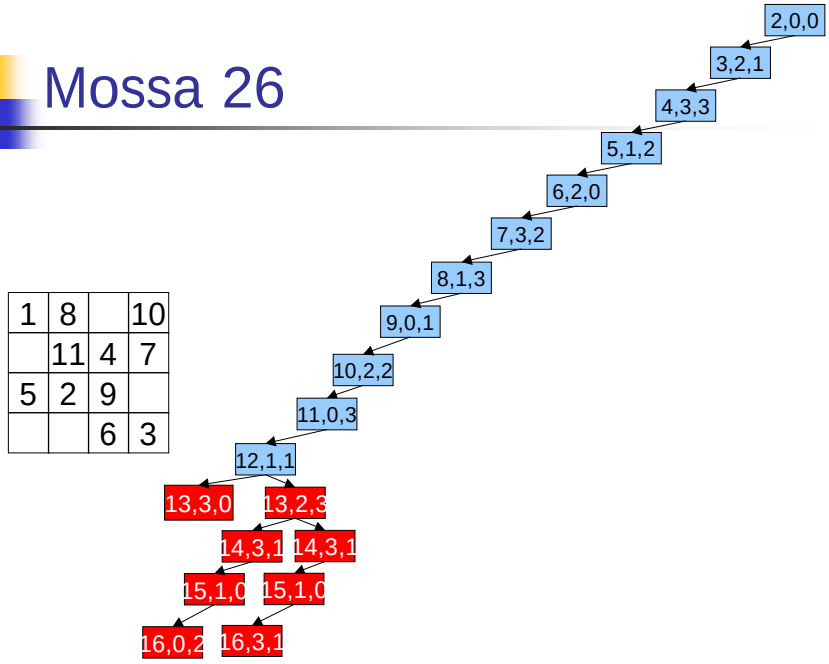
Mossa 25

1	8		10
	11	4	7
5	2	9	12
		6	3



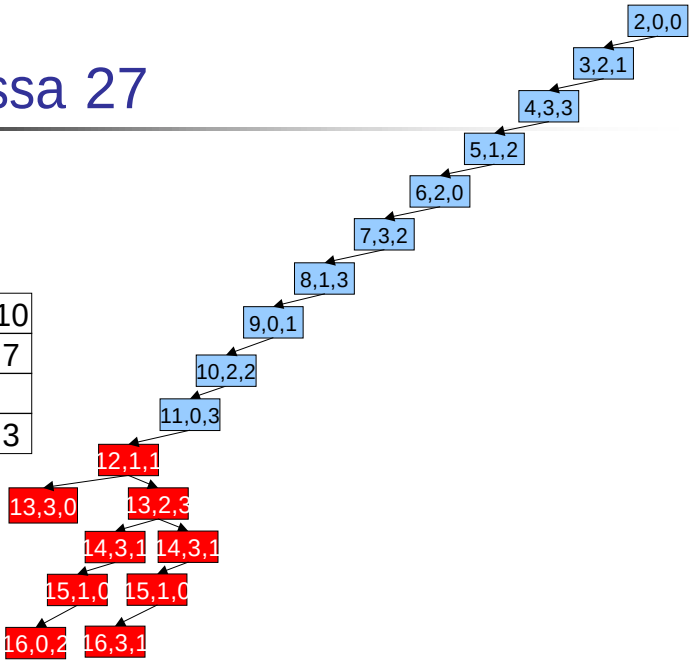
Mossa 26

1	8		10
	11	4	7
5	2	9	
		6	3



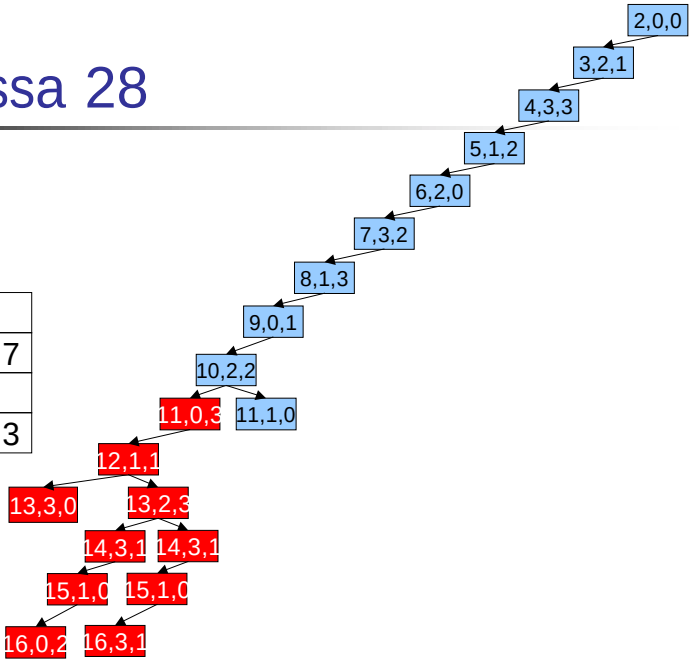
Mossa 27

1	8		10
		4	7
5	2	9	
		6	3



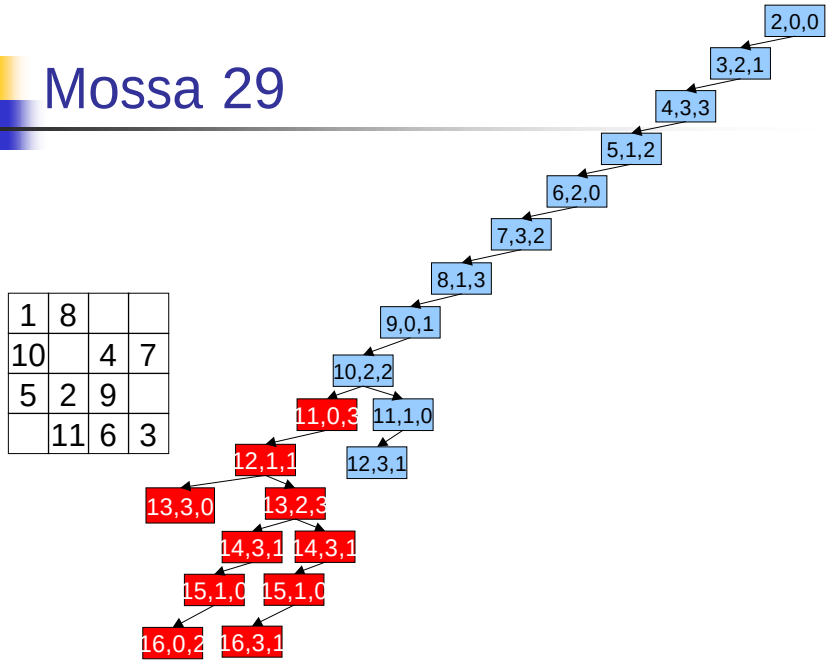
Mossa 28

1	8		
10		4	7
5	2	9	
		6	3



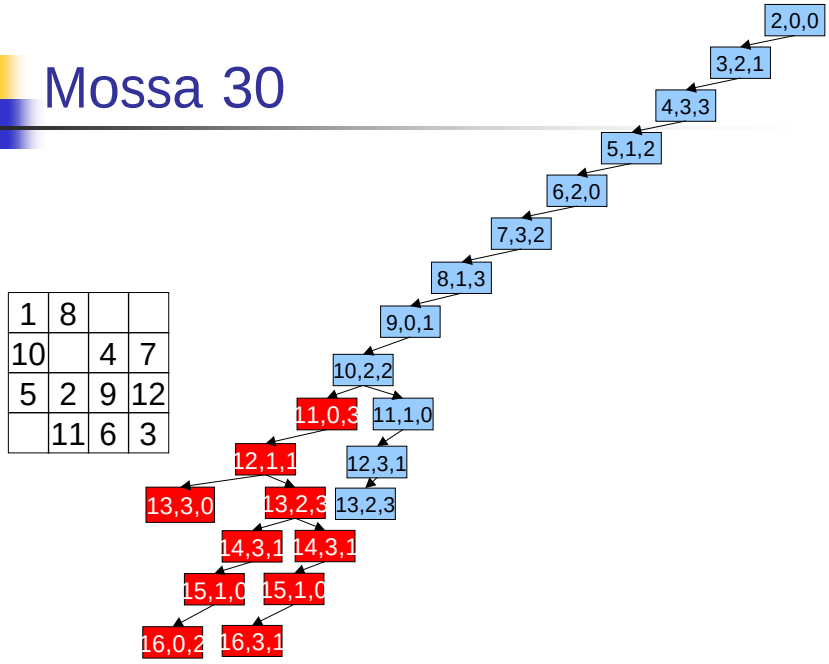
Mossa 29

1	8		
10		4	7
5	2	9	
	11	6	3



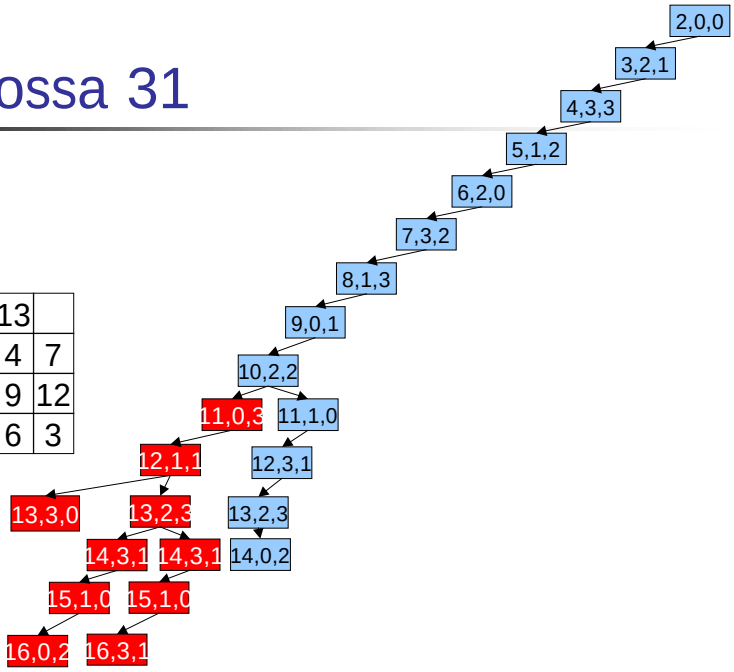
Mossa 30

1	8		
10		4	7
5	2	9	12
	11	6	3



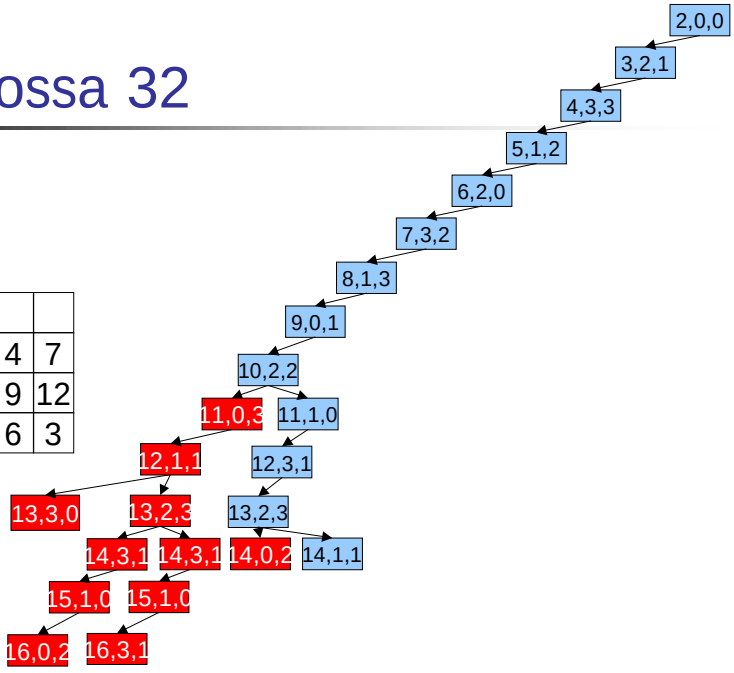
Mossa 31

1	8	13	
10		4	7
5	2	9	12
	11	6	3



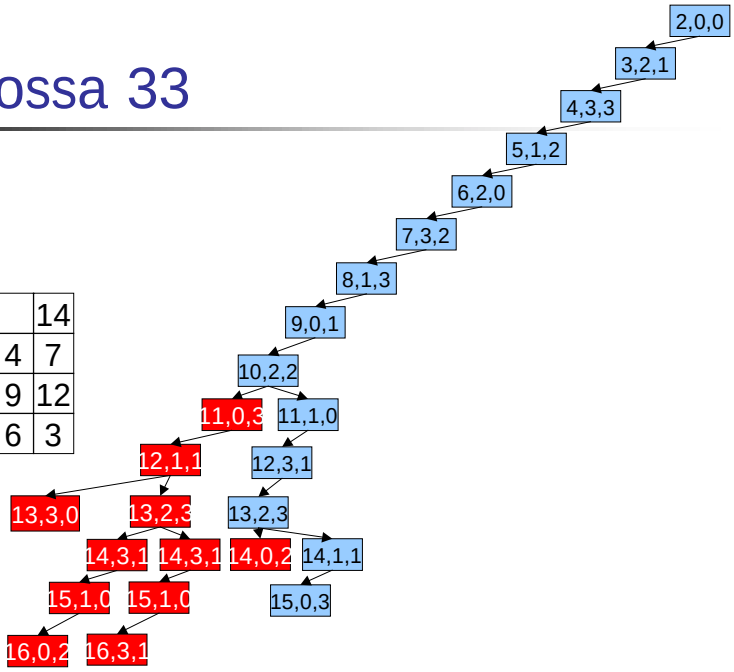
Mossa 32

1	8		
10	13	4	7
5	2	9	12
	11	6	3



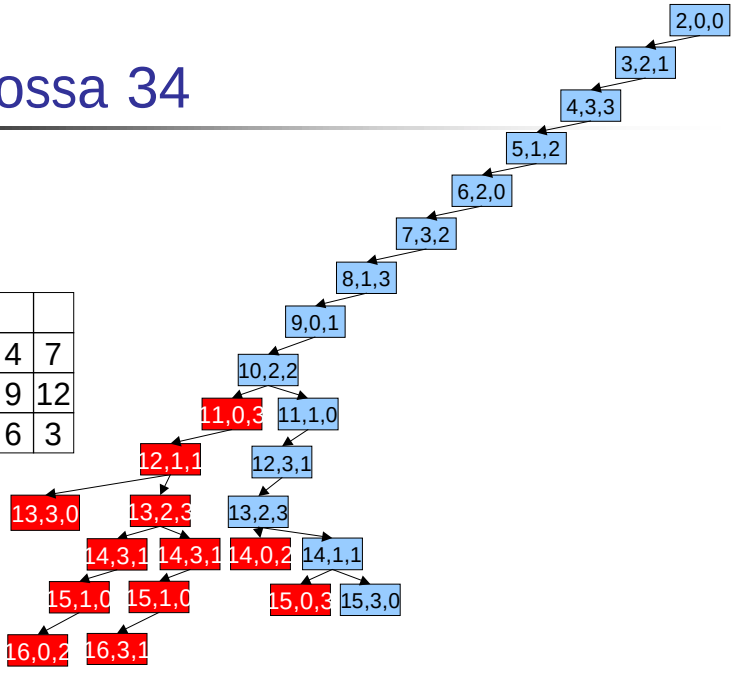
Mossa 33

1	8		14
10	13	4	7
5	2	9	12
	11	6	3



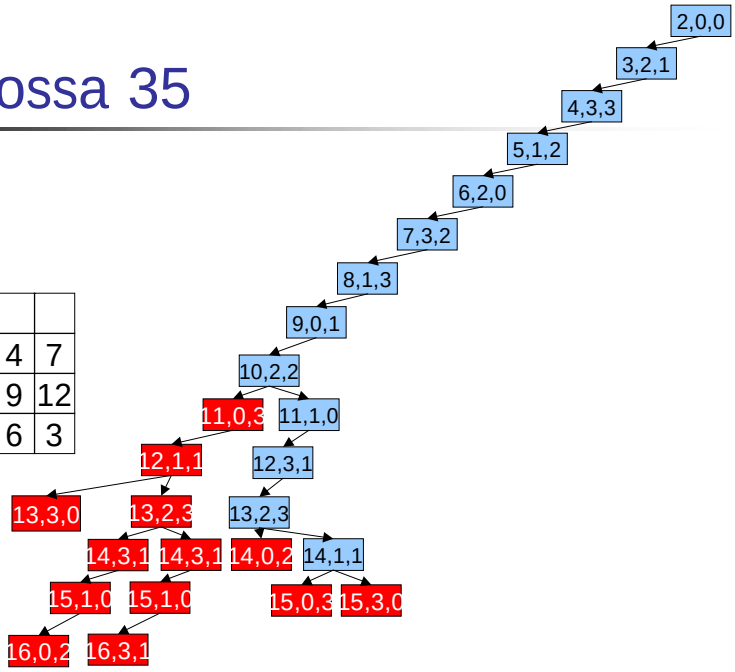
Mossa 34

1	8		
10	13	4	7
5	2	9	12
14	11	6	3



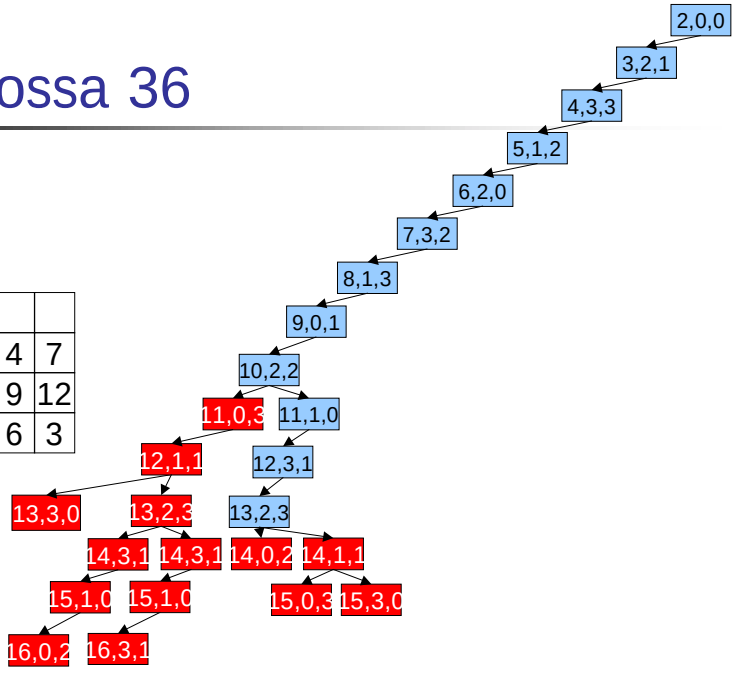
Mossa 35

1	8		
10	13	4	7
5	2	9	12
	11	6	3



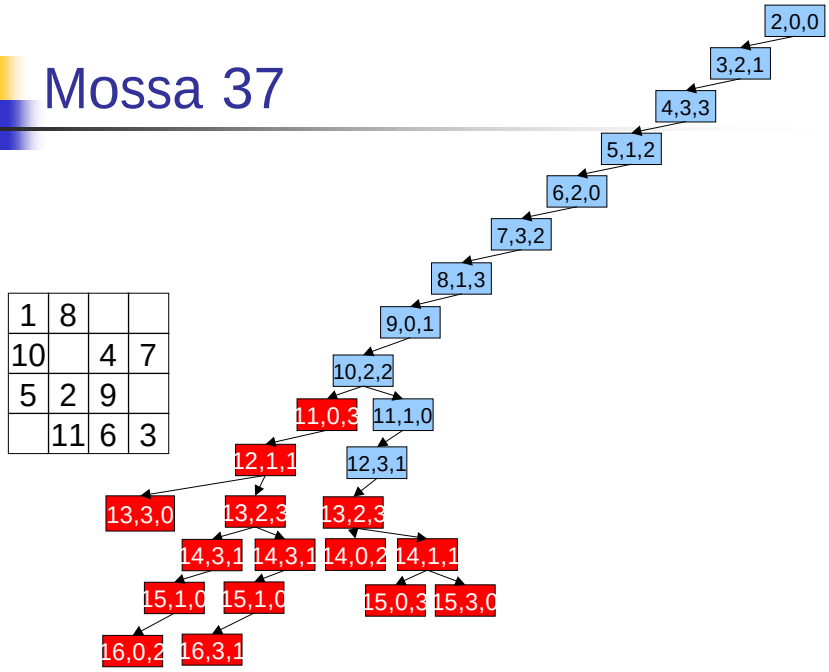
Mossa 36

1	8		
10		4	7
5	2	9	12
	11	6	3



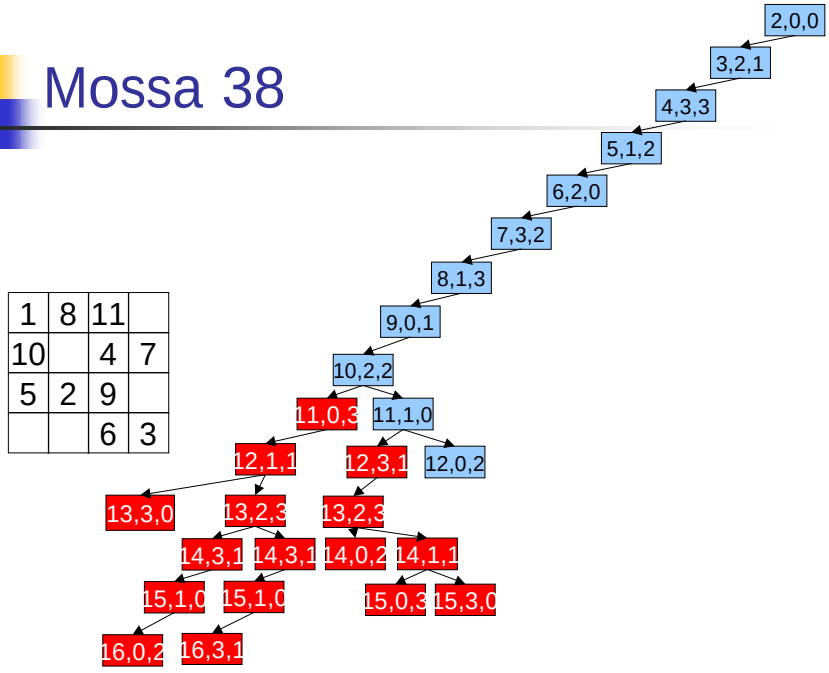
Mossa 37

1	8		
10		4	7
5	2	9	
	11	6	3



Mossa 38

1	8	11	
10		4	7
5	2	9	
		6	3



Complessità

- Il numero di mosse possibili ad ogni passo è al più 8.
- Il numero di passi è N^2.
- Quindi l'albero ha un numero di nodi $\leq 8^{N\hat{}2}$.
- Nel caso peggiore
  - la soluzione corrisponde alla foglia più a destra
  - l'albero è completo.
- In tal caso il numero di chiamate recursive prima di trovare la soluzione è $\Theta(8^{N\hat{}2})$.