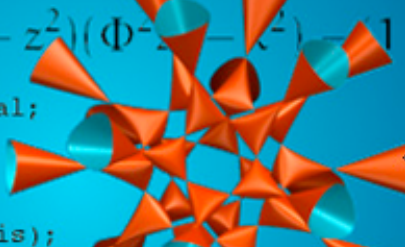


Paralleles Radix Sort

Seminar „Parallelverarbeitung und
Grafikhardware“, SS 2010

Betreuer: Nicolas Menzel

```
4(\Phi^2x^2 - y^2)(\Phi^2y^2 - z^2)(\Phi^2z^2 - x^2) - 1  
perspective=central;  
spec_p=150.0;  
radius=10.0;  
sextic=rotate(  
    sextic,-0.1,xAxis);
```



Übersicht

- Effizientester Algorithmus für kleine Schlüssel
 - Laufzeit **$O(n)$** für feste Länge
- Basiert auf dem *Counting Sort*
- *Counting Sort* reduzierbar
 - Parallele Präfixsumme oder *Scan*
- 23% schneller als effizientester CPU Multicore Algorithmus
- Implementierung in CUDA

Radix Sort

- Sortierschlüssel aus Zeichen eines endlichen Alphabets
 - z.B: Ziffern 0-9
- Zwei Phasen
 - *Partitionierungsphase*
 - Teile Daten in Fächer auf
 - *Sammelphase*
 - Entnehme den Fächern Daten

Beispiel

Eingabewerte

124, 523, 483, 128, 923, 584

Partitionierungsphase

0	1	2	3	4	5	6	7	8	9
			523	124				128	
			483	584					
			923						

Sammelphase

523, 483, 923, 124, 584, 128