RÄKNESORTERING

COUNTING SORT(V[1..n], f: ELEMENT → [1..k])=

HJÄLPARRAY FÖR RÄKNING: C[1..k]

HJÄLPARRAY FÖR LAGRING AV RESULTATET: RES[1..n]

```
FOR i←1 TO k DO C[i]←0
FOR j←1 TO n DO C[f(V[j])]++
sum←0
FOR i←k DOWNTO 1 DO
    sum←sum+C[i]
    C[i]← n−sum+1
FOR j←1 TO n DO
    RES[C[f(V[j])]] ← V[j]
    C[f(V[j])]++
RETURN RES
```

TIDSKOMPLEXITET: $\Theta(k+n)$

EXEMPEL:

V[1..10] = | 17 | 2 | 2 | 4711 | 2 | 17 | 17 | 4711 | 2 | 17 |

f(2)=1 , f(17)=2, f(4711)=3

C[1..3] EFTER RÄKNESLINGAN = | 4 | 4 | 2 |

C[1..3] EFTER INDEXOMRÄKNING= | 1 | 5 | 9 |

RES[1..10]= | 2 2 2 2 | 17 17 17 17 | 4711 4711 |

NÅGRA FALL DÅ MAN KAN SORTERA I $o(n \log n)$

1. BARA ETT KONSTANT ANTAL <u>OLIKA</u> ELEMENT SKA SORTERAS.

   $\Theta(n)$ MED RÄKNESORTERING (COUNTING SORT)

2. ELEMENTEN SOM SKA SORTERAS ÄR TAL SOM ÄR JÄMNT FÖRDELADE I ETT VISST INTERVALL.

   $\Theta(n)$ MED BUCKETSORT

3. ELEMENTEN SOM SKA SORTERAS ÄR STRÄNGAR SOM BESTÅR AV d "SIFFROR". $(v[i] = s_{i,1}\, s_{i,2} \ldots s_{i,d})$

   $\Theta(nd)$ MED RADIXSORTERING:

   ```
   FOR i ← d DOWNTO 1 DO
       SORTERA v[1..n] EFTER SIFFRA i MED
       EN STABIL SORTERINGSALGORITM
   ```

   - OM d ÄR KONSTANT FÅR VI LINJÄR TIDSKOMPLEXITET.
   - OM VI RÄKNAR ANTALET SIFFROR I INDATA FÅR VI LINJÄR TIDSKOMPLEXITET $\Theta(N)$ DÄR $N = nd$.

Radixsorteringsexempel med d=3

Osorterat:	Pass 1:	Pass 2:	Pass 3:
480	480	902	009
973	381	905	381
902	902	009	419
905	532	816	480
532	652	419	532
652	973	532	652
783	783	652	653
009	653	653	783
653	905	973	816
419	816	480	902
816	009	381	905
381	419	783	973