

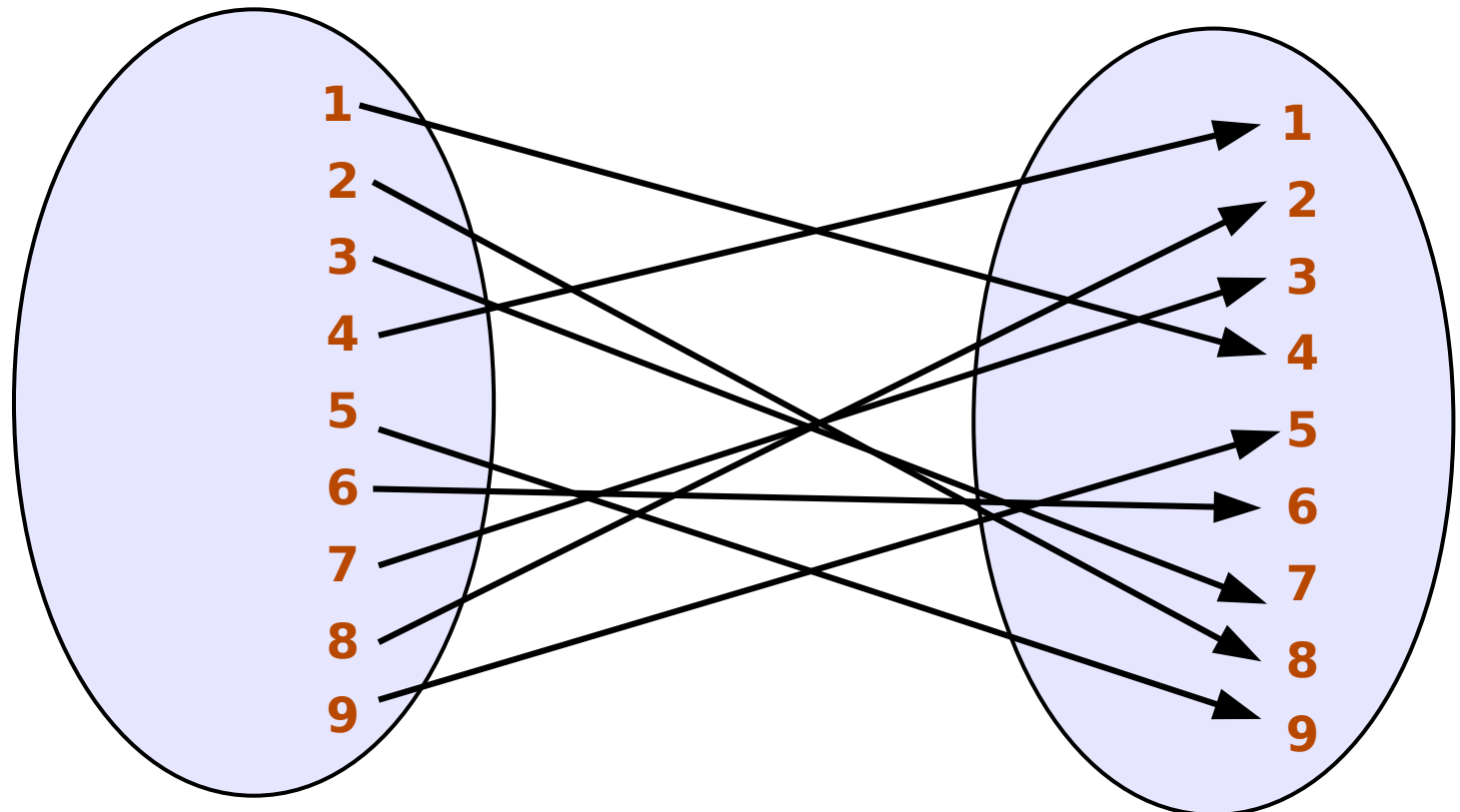


Sorteren en Permuteren

Permutatie

Definitie: een permutatie is een bijectie waarvan het domein en het bereik dezelfde verzameling zijn.

Voorbeeld:





Sorteren en Permuteren

Cykelnotatie

Algoritme

- 1) Maak een cykel van het eerste element
- 2) Voeg het beeld van het laatste element toe als dat nog niet in de cykel zit
- 3) Herhaal vanaf stap 2 totdat je bij het eerste element van de cykel bent
- 4) Als je alle elementen hebt gebruikt ben je klaar
- 5) Anders maak je een cykel van een resterend element
- 6) En gaat door bij stap 2

Voorbeeld: $(4,8,7,1,9,6,3,2,5) = (1\ 4)(2\ 8)(3\ 7)(5\ 9)(6)$

Rekentijd: recht evenredig met aantal elementen



Sorteren en Permuteren

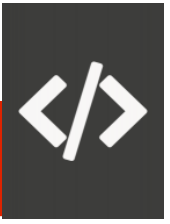
Stelling

Elke permutatie van een eindige verzameling kan worden samengesteld uit ten hoogste $\frac{1}{2} \times N \times (N+1)$ verwisselingen van twee naburige elementen, dus (1 2), (2 3), (3 4), ..., (N-1 N).

- de grootste afstand tussen (1,2,...,N) en (N,N-1,...,2,1).
- Met (1 2)(2 3)(3 4)...(N-1 N) verplaats je de 1 naar de laatste positie; het is gelijk aan (N N-1.....3 2 1).
- dan staat de 2 vooraan; die verplaats je naar de éénnalaatste plaats met (1 2)(3 4)(5 6)...(N-2 N-1), enz.
- het aantal verwisselingen komt daarmee op

$$\sum_{i=0}^N i = \frac{1}{2} \times N \times (N+1)$$





Sorteren en Permuteren

Stelling

Elke permutatie van N elementen kan worden geschreven als een combinatie van maximaal N verwisselingen van twee elementen.

- $N = 4$: maximaal 3 verwisselingen (alle mogelijkheden controleren)
- $N+1$: neem aan dat de stelling klopt voor alle $x \leq N$. Dan is elke permutatie van $N+1$ elementen te maken door $(1,2,3,\dots,N)$ te permuteren in N stappen tot (a,b,c,\dots,z) , wat we uitbreiden tot $(a,b,c,\dots,z,N+1)$, wat door 1 verwisseling alle permutaties van de vormen $(N+1,a,b,c,\dots,z)$, $(a,N+1,b,c,\dots,z)$, $(a,b,N+1,c,\dots,z)$, t/m $(a,b,c,\dots,N+1,z)$ oplevert; de permutatie $(a,b,c,\dots,z,N+1)$ vergt 0 extra verwisselingen.





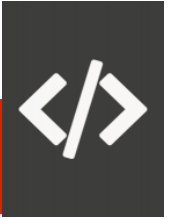
Sorteren en Permuteren

Parallelliseren

Op een massief parallel computer kan het vinden van de cykels sneller.

- Splits cykels van lengte N op in twee cykels van $N/2$.
- Doe permutaties van beide cykels parallel.
- Doe in 1 parallelle stap maximaal $N/4$ verwisselingen van 2 elementen uit verschillende deelcykels.

Rekentijd: in de orde van $2 \log(N)$ verwisselingen als je N processoren gebruikt.



Sorteren en Permuteren

Metasort

Je kunt de taken van een sorteerprogramma in tweeën verdelen: het eerste deel analyseert de invoer en genereert de code voor het tweede gedeelte, dat de invoer op de gewenste manier permuteert.

Voordeel: als je dezelfde file meer dan eens wilt sorteren, hoef je alleen de permutatie-fase te herhalen. Zoals we zagen kan dat laatste in lineaire tijd worden uitgevoerd.

De analyse fase zou daaruit kunnen bestaan dat voor elk getal in de invoer wordt gezocht welke positie het moet krijgen in de uitvoer. Dat kost dan $O(N^2)$ rekenstappen.



Sorteren en Permuteren

Algoritme om invoer te analyseren

Maak een lege binaire boom $B = \perp$; doe voor elk getal G :

1) zet $K \leftarrow L \leftarrow B \rightarrow \text{root}$

2) als $K = \perp$

1) zet $L \leftarrow \text{knoop}(G)$; $L \rightarrow \text{count} \leftarrow 0$

2) ga verder met volgende getal vanaf stap 1.

3) als $K \neq \perp$

1) als $G = K \rightarrow \text{key}$

1) zet $K \rightarrow \text{key} \leftarrow K \rightarrow \text{key} + 1$

2) ga verder met het volgende getal vanaf stap 1.

2) als $G < K \rightarrow \text{key}$

1) zet $L \leftarrow K$; $K = K \rightarrow \text{left}$

2) ga verder met dit getal vanaf stap 2.

3) als $G > K \rightarrow \text{key}$

1) zet $L \leftarrow K$; $K = K \rightarrow \text{right}$

2) ga verder met dit getal vanaf stap 2.



Sorteren en Permuteren

Bespreking

Dit model komt neer op het doorlopen van een beslisboom, wat voor elk getal $O(2^{\log(N)})$ stappen kost. Je kunt op basis hiervan een sorteerprogramma ontwerpen dat $O(N \times 2^{\log(N)})$ stappen vergt. Het is tevens mogelijk om alle getallen parallel te verwerken.

Het Quicksort algoritme volgt ongeveer dit model, maar alle getallen worden telkens aan 1 test onderworpen en eventueel verwisseld. Het Mergesort algoritme komt overeen met dit model. Een algoritme dat het model van dia 6 volgt is Selectionsort en werkt minder efficiënt. Bubblesort is een algoritme dat telkens buren vergelijkt en eventueel verwisselt. Cyclesort maakt gebruik van het algoritme van dia 2; het is kost $O(N^2)$ stappen, maar slechts $O(N)$ verplaatsingen.

Op de volgende dia een voorbeeld van code om permutaties te genereren die is gewijzigd zodat een sorteerprogramma ontstaat.



Sorteren en Permuteren

Sorteren door permuteren in C

```
/* recursieve sorteer functie */
void loop1( int i, int j)
{
    if( i > 0)
    {
        if( j >= i)
        {
            if( perm[j-1] <= perm[i-1])
            {
                swap( i - 1, j - 1);
                loop1( i-1, N);
            }
            loop1( i, j-1);
        }
    }
}
```



Sorteren en Permuteren

Kan het nog beter?

- Bedenk dat veel processoren nu 64 bits tegelijk kunnen verwerken en dat veel algoritmen arrays sorteren, zodat een plaatsje vinden voor een getal een bottle-neck vormt. Volgend algoritme wordt Bucketsort genoemd.
- Je vindt in lineaire tijd het grootste en kleinste getal.
- Vervolgens kun je een array maken van N *linked lists*.
- Uit elk getal bereken je een index en voegt het toe aan die lijst.
- Vervolgens loop je het array af: is de lijst leeg, doe je niets, zijn de grootste en kleinste elementen gelijk, dan druk je alles af, en anders moet je de (veel kleinere) lijst verder sorteren, bijvoorbeeld met recursie.

einde van dit onderwerp



Sorteren en Permuteren

Bronvermelding

Deze presentatie bevat informatie ontleend aan de volgende bronnen:

- <https://nl.wikipedia.org/wiki/Permutatie>
- <https://nl.wikipedia.org/wiki/Cykelnotatie>
- https://en.wikipedia.org/wiki/Sorting_algorithm
- https://en.wikipedia.org/wiki/Cycle_sort
- <http://bakunin.xs4all.nl/artikelen/permutatie.pdf>
- <http://bakunin.xs4all.nl/artikelen/permutoeders.pdf>