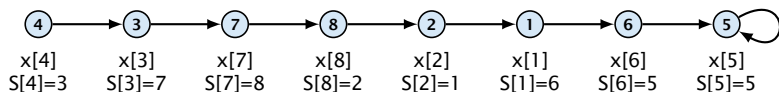


Parallel Prefix

Input: a linked list given by successor pointers; a value $x[i]$ for every list element; an operator $*$;

Output: for every list position ℓ the sum (w.r.t. $*$) of elements after ℓ in the list (including ℓ)



Parallel Prefix

Algorithm 7 ParallelPrefix

```
1: for  $1 \leq i \leq n$  pardo
2:    $P[i] \leftarrow S[i]$ 
3:   while  $S[i] \neq S[S[i]]$  do
4:      $x[i] \leftarrow x[i] * x[S[i]]$ 
5:      $S[i] \leftarrow S[S[i]]$ 
6:   if  $P[i] \neq i$  then  $x[i] \leftarrow x[i] * x[S(i)]$ 
```

The algorithm runs in time $\mathcal{O}(\log n)$.

It has work requirement $\mathcal{O}(n \log n)$. non-optimal

This technique is also known as pointer jumping

Parallel Prefix

Algorithm 7 ParallelPrefix

```
1: for  $1 \leq i \leq n$  pardo
2:    $P[i] \leftarrow S[i]$ 
3:   while  $S[i] \neq S[S[i]]$  do
4:      $x[i] \leftarrow x[i] * x[S[i]]$ 
5:      $S[i] \leftarrow S[S[i]]$ 
6:   if  $P[i] \neq i$  then  $x[i] \leftarrow x[i] * x[S(i)]$ 
```

The algorithm runs in time $\mathcal{O}(\log n)$.

It has work requirement $\mathcal{O}(n \log n)$. non-optimal

This technique is also known as pointer jumping

Parallel Prefix

Algorithm 7 ParallelPrefix

```
1: for  $1 \leq i \leq n$  pardo
2:    $P[i] \leftarrow S[i]$ 
3:   while  $S[i] \neq S[S[i]]$  do
4:      $x[i] \leftarrow x[i] * x[S[i]]$ 
5:      $S[i] \leftarrow S[S[i]]$ 
6:   if  $P[i] \neq i$  then  $x[i] \leftarrow x[i] * x[S(i)]$ 
```

The algorithm runs in time $\mathcal{O}(\log n)$.

It has work requirement $\mathcal{O}(n \log n)$. **non-optimal**

This technique is also known as **pointer jumping**

Parallel Prefix

Algorithm 7 ParallelPrefix

```
1: for  $1 \leq i \leq n$  pardo
2:    $P[i] \leftarrow S[i]$ 
3:   while  $S[i] \neq S[S[i]]$  do
4:      $x[i] \leftarrow x[i] * x[S[i]]$ 
5:      $S[i] \leftarrow S[S[i]]$ 
6:   if  $P[i] \neq i$  then  $x[i] \leftarrow x[i] * x[S(i)]$ 
```

The algorithm runs in time $\mathcal{O}(\log n)$.

It has work requirement $\mathcal{O}(n \log n)$. **non-optimal**

This technique is also known as **pointer jumping**