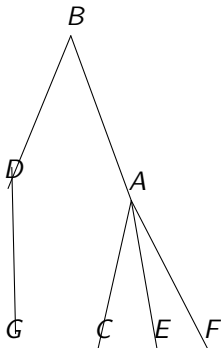
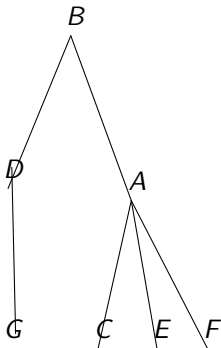


Traversée d'une arborescence

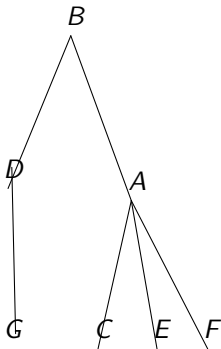


```
1  $R.\text{numPrefixe} \leftarrow \text{pref}$  ;  
    $\text{pref} ++$  ;  
   pour chacun des successeurs S de R faire  
2   |  $(\text{pref}, \text{post}) \leftarrow$   
     |    $\text{TraverseeRecursive} (S, (\text{pref}, \text{post}))$ ;  
   |  
   fin  
3  $R.\text{numPostfixe} \leftarrow \text{post}$  ;  $\text{post} ++$  ;  
   retourner  $(\text{pref}, \text{post})$ 
```



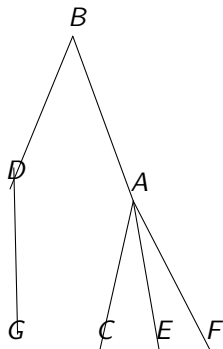
```
1  $R.numPrefix \leftarrow pref$  ;  
    $pref ++$  ;  
   pour chacun des successeurs S de R faire  
2   |  $(pref, post) \leftarrow$   
   |    $TraverseeRecursive (S, (pref, post))$ ;  
   |  
   fin  
3  $R.numPostfix \leftarrow post$  ;  $post ++$  ;  
   retourner  $(pref, post)$ 
```

On identifiera une arborescence avec le nom de sa racine



```
1  $R.numPrefix \leftarrow pref$  ;  
    $pref ++$  ;  
   pour chacun des successeurs S de R faire  
2   |  $(pref, post) \leftarrow$   
   |    $TraverseeRecursive (S, (pref, post))$ ;  
   |  
   fin  
3  $R.numPostfix \leftarrow post$  ;  $post ++$  ;  
   retourner  $(pref, post)$ 
```

On identifiera une arborescence avec le nom de sa racine
Ainsi, l'appel initial sera $TR(B, 1, 1)$



```

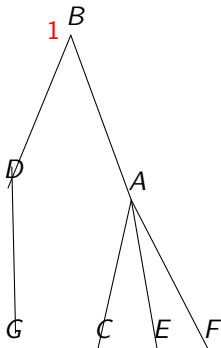
1  $R.numPrefix \leftarrow pref$  ;
   $pref ++$  ;
  pour chacun des successeurs S de R faire
2    $(pref, post) \leftarrow$ 
      $TraverseeRecursive (S, (pref, post))$ ;
  fin
3  $R.numPostfix \leftarrow post$  ;  $post ++$  ;
  retourner  $(pref, post)$ 

```

Ainsi, l'appel initial sera $TR(B, 1, 1)$

R B pref 1 post 1

B

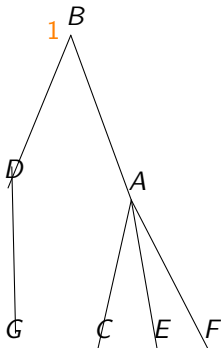


```

1  R.numPrefix  $\leftarrow$  pref ;
   pref ++ ;
   pour chacun des successeurs S de R faire
2  |   (pref,post)  $\leftarrow$ 
   |       TraverseeRecursive (S,(pref,post));
   |
   fin
3  R.numPostfixe  $\leftarrow$  post ; post ++ ;
   retourner (pref,post)
  
```

R B *pref* 1 *post* 1

B

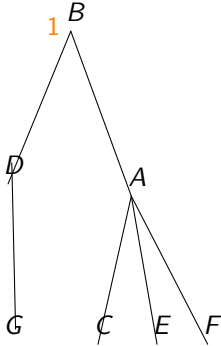


```

1 R.numPrefix ← pref ;
  pref ++;
  pour chacun des successeurs S de R faire
2   (pref,post) ←
    TraverseeRecursive (S,(pref,post));
  fin
3 R.numPostfixe ← post ; post ++;
  retourner (pref,post)
  
```

R B pref 2 post 1

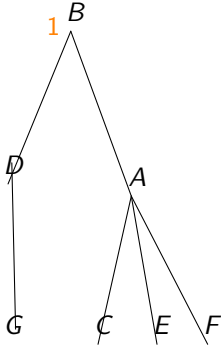
B



```
1  $R.\text{numPrefixe} \leftarrow \text{pref}$  ;  
    $\text{pref} ++$  ;  
   pour chacun des successeurs S de R faire  
2   |  $(\text{pref}, \text{post}) \leftarrow$   
     |  $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$  ;  
   | fin  
3  $R.\text{numPostfixe} \leftarrow \text{post}$  ;  $\text{post} ++$  ;  
   retourner  $(\text{pref}, \text{post})$ 
```

R B pref 2 post 1

B

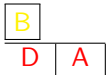


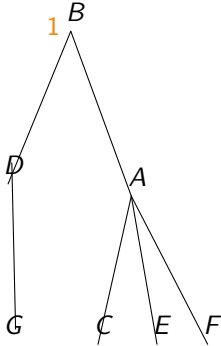
```

1  $R.numPrefix \leftarrow pref$  ;
    $pref ++$  ;
   pour chacun des successeurs S de R faire
2    $(pref, post) \leftarrow$ 
      $TraverseeRecursive(S, (pref, post))$ ;
   fin
3  $R.numPostfix \leftarrow post$  ;  $post ++$  ;
   retourner  $(pref, post)$ 

```

R B pref 2 post 1

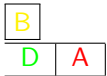


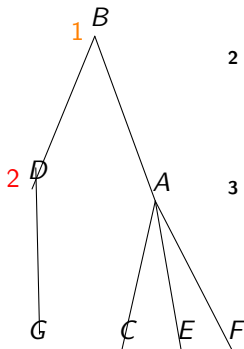


```

1  $R.\text{numPrefixe} \leftarrow \text{pref}$  ;
   $\text{pref}++$  ;
  pour chacun des successeurs S de R faire
2    $(\text{pref}, \text{post}) \leftarrow$ 
     TraverseeRecursive (S, (pref, post));
  fin
3  $R.\text{numPostfixe} \leftarrow \text{post}$  ;  $\text{post}++$  ;
  retourner (pref, post)
  
```

R D pref 2 post 1

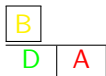


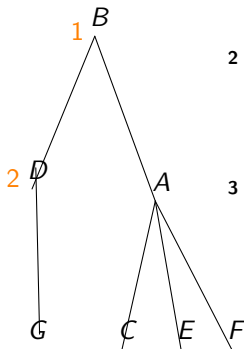


```

1  R.numPrefix  $\leftarrow$  pref ;
   pref ++ ;
   pour chacun des successeurs S de R faire
2  |   (pref,post)  $\leftarrow$ 
   |       TraverseeRecursive (S,(pref,post));
   |
   fin
3  R.numPostfixe  $\leftarrow$  post ; post ++ ;
   retourner (pref,post)
  
```

R D *pref* 2 *post* 1



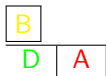


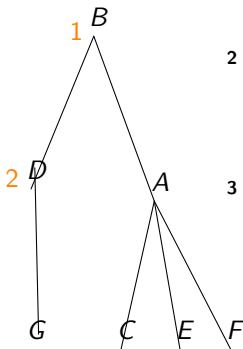
```

1  $R.\text{numPrefix} \leftarrow \text{pref}$  ;
    $\text{pref}++$  ;
   pour chacun des successeurs S de R faire
2    $(\text{pref}, \text{post}) \leftarrow$ 
      $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$  ;
   fin
3  $R.\text{numPostfix} \leftarrow \text{post}$  ;  $\text{post}++$  ;
   retourner  $(\text{pref}, \text{post})$ 

```

R D pref 3 post 1



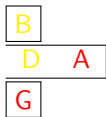


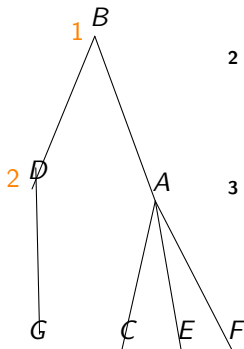
```

1  $R.numPrefix \leftarrow pref$  ;
   $pref++$  ;
  pour chacun des successeurs  $S$  de  $R$  faire
2    $(pref, post) \leftarrow$ 
      $TraverseeRecursive(S, (pref, post))$  ;
  fin
3  $R.numPostfix \leftarrow post$  ;  $post++$  ;
  retourner  $(pref, post)$ 

```

R G pref 3 post 1



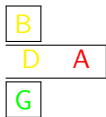


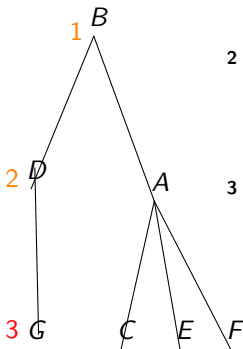
```

1  $R.\text{numPrefix} \leftarrow \text{pref}$  ;
   $\text{pref}++$  ;
  pour chacun des successeurs  $S$  de  $R$  faire
2    $(\text{pref}, \text{post}) \leftarrow$ 
     TraverseeRecursive ( $S, (\text{pref}, \text{post})$ );
  fin
3  $R.\text{numPostfixe} \leftarrow \text{post}$  ;  $\text{post}++$  ;
  retourner ( $\text{pref}, \text{post}$ )

```

R G pref 3 post 1

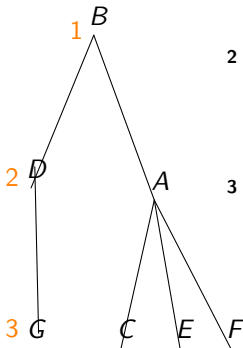




```

1  R.numPrefix  $\leftarrow$  pref ;
   pref ++ ;
   pour chacun des successeurs S de R faire
2  |   (pref, post)  $\leftarrow$ 
      TraverseeRecursive (S, (pref, post));
   |
   fin
3  R.numPostfixe  $\leftarrow$  post ; post ++ ;
   retourner (pref, post)

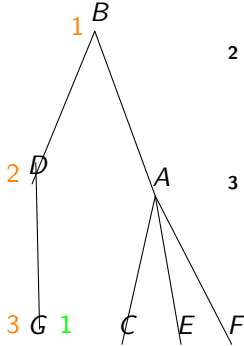
```



```

1 R.numPrefixe  $\leftarrow$  pref ;
   pref ++;
   pour chacun des successeurs S de R faire
2   | (pref,post)  $\leftarrow$ 
      TraverseeRecursive (S,(pref,post));
   |
fin
3 R.numPostfixe  $\leftarrow$  post ; post ++;
   retourner (pref,post)

```

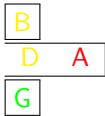



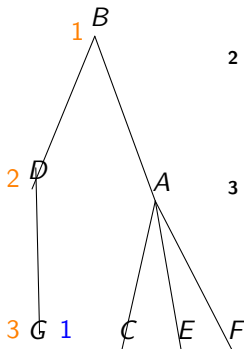
```

1  $R.\text{numPrefixe} \leftarrow \text{pref}$  ;
   $\text{pref}++$  ;
  pour chacun des successeurs  $S$  de  $R$  faire
2    $(\text{pref}, \text{post}) \leftarrow$ 
      $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$ ;
  fin
3  $R.\text{numPostfixe} \leftarrow \text{post}$  ;  $\text{post}++$  ;
  retourner  $(\text{pref}, \text{post})$ 

```

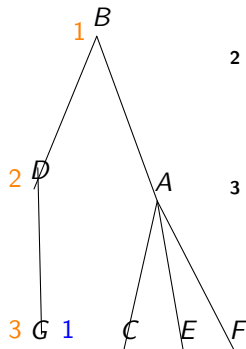
R G pref 4 post 1



R

 pref

 post

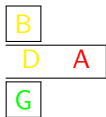


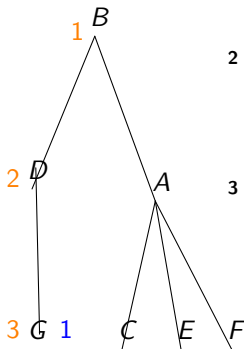
```

1  $R.\text{numPrefix} \leftarrow \text{pref}$  ;
   $\text{pref}++$  ;
  pour chacun des successeurs S de R faire
2    $(\text{pref}, \text{post}) \leftarrow$ 
      $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$  ;
  fin
3  $R.\text{numPostfix} \leftarrow \text{post}$  ;  $\text{post}++$  ;
  retourner  $(\text{pref}, \text{post})$ 

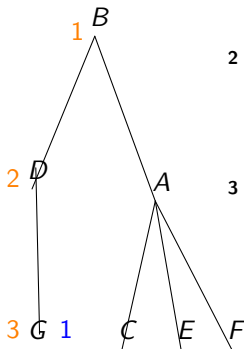
```

R G pref 4 post 2

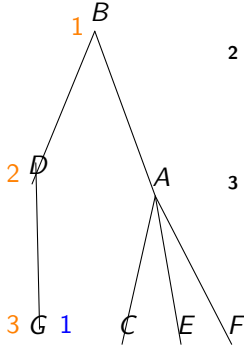




R G pref 4 post 2



R G pref 4 post 2



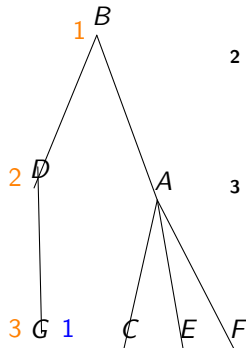
```

1 R.numPrefix ← pref ;
  pref ++ ;
  pour chacun des successeurs S de R faire
2   (pref,post) ←
    TraverseeRecursive (S,(pref,post));
  fin
3 R.numPostfix ← post ; post ++ ;
  retourner (pref,post)
  
```

R G pref 4 post 2



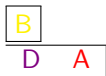
On sort de l'appel TraverseeRecursive (G,3,1) *pref*, *post* "changent"



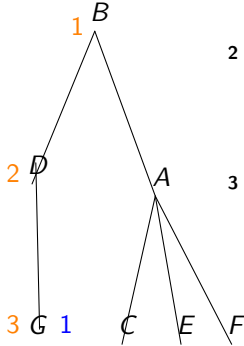
```

1 R.numPrefix ← pref ;
  pref ++ ;
  pour chacun des successeurs S de R faire
2   (pref,post) ←
    TraverseeRecursive (S,(pref,post));
  fin
3 R.numPostfix ← post ; post ++ ;
  retourner (pref,post)
  
```

R G pref 4 post 2



On sort de l'appel TraverseeRecursive (G,3,1)
donc de la répétitive

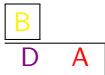


```

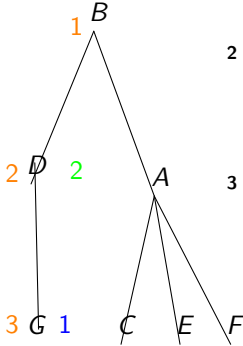
1 R.numPrefix ← pref ;
  pref ++ ;
  pour chacun des successeurs S de R faire
2   (pref,post) ←
     TraverseeRecursive (S,(pref,post));
  fin
3 R.numPostfix ← post ; post ++ ;
  retourner (pref,post)

```

R D pref 4 post 2



On sort de l'appel TraverseeRecursive (G,3,1)
donc de la répétitive **avec R=D**

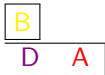


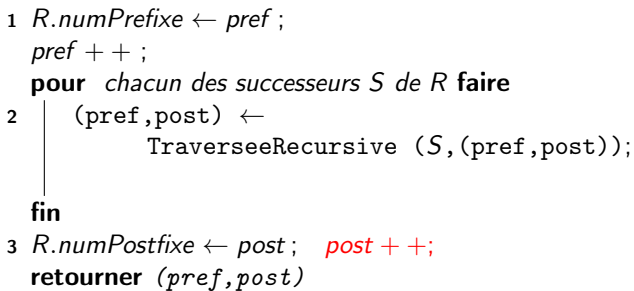
```

1  R.numPrefixe  $\leftarrow$  pref ;
   pref ++ ;
   pour chacun des successeurs S de R faire
2  |   (pref,post)  $\leftarrow$ 
      TraverseeRecursive (S,(pref,post));
   |
   fin
3  R.numPostfixe  $\leftarrow$  post ; post ++ ;
   retourner (pref,post)

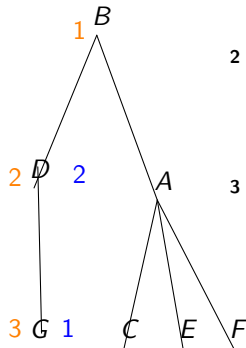
```

R D pref 4 post 2





B
D A

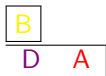


```

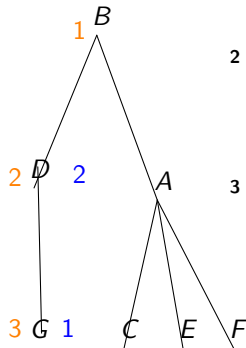
1   $R.\text{numPrefix} \leftarrow \text{pref}$  ;
    $\text{pref} ++$  ;
   pour chacun des successeurs S de R faire
2  |    $(\text{pref}, \text{post}) \leftarrow$ 
      |    $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$ ;
   fin
3   $R.\text{numPostfix} \leftarrow \text{post}$  ;  $\text{post} ++$  ;
   retourner  $(\text{pref}, \text{post})$ 

```

R D pref 4 post 3



On sort de l'appel $\text{TraverseeRecursive}(D, 2, 1)$



```

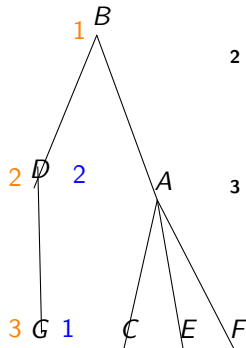
1   $R.\text{numPrefix} \leftarrow \text{pref}$  ;
    $\text{pref}++$  ;
   pour chacun des successeurs  $S$  de  $R$  faire
2  |    $(\text{pref}, \text{post}) \leftarrow$ 
      |    $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$ ;
   fin
3   $R.\text{numPostfix} \leftarrow \text{post}$  ;  $\text{post}++$  ;
   retourner  $(\text{pref}, \text{post})$ 

```

R D pref 4 post 3



On sort de l'appel $\text{TraverseeRecursive}(D, 2, 1)$
 donc on continue la répétitive



```

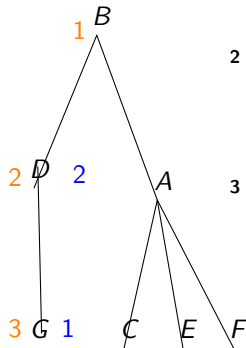
1   $R.\text{numPrefix} \leftarrow \text{pref}$  ;
    $\text{pref}++$  ;
   pour chacun des successeurs  $S$  de  $R$  faire
2  |    $(\text{pref}, \text{post}) \leftarrow$ 
      |    $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$ ;
   fin
3   $R.\text{numPostfix} \leftarrow \text{post}$  ;  $\text{post}++$  ;
   retourner  $(\text{pref}, \text{post})$ 

```

R D pref 4 post 3



On sort de l'appel $\text{TraverseeRecursive}(D, 2, 1)$
 donc on continue la répétitive **avec** $S=A$

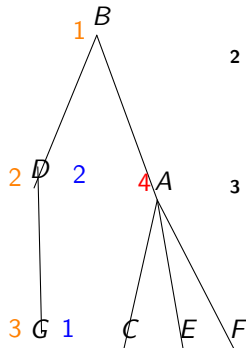


```

1 R.numPrefix ← pref ;
  pref ++ ;
  pour chacun des successeurs S de R faire
2   (pref,post) ←
    TraverseeRecursive (S,(pref,post));
  fin
3 R.numPostfixe ← post ; post ++ ;
  retourner (pref,post)
  
```

R A pref 4 post 3





```

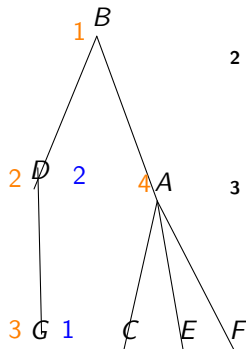
1  R.numPrefix  $\leftarrow$  pref ;
   pref ++ ;
   pour chacun des successeurs S de R faire
2  |   (pref,post)  $\leftarrow$ 
      |       TraverseeRecursive (S,(pref,post));
   |
   fin
3  R.numPostfixe  $\leftarrow$  post ; post ++ ;
   retourner (pref,post)

```

R A pref 4 post 3

B

A



```

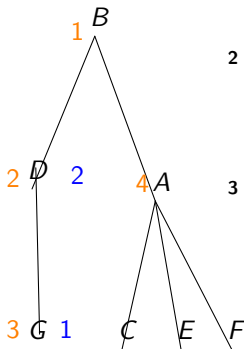
1  $R.\text{numPrefix} \leftarrow \text{pref}$  ;
    $\text{pref}++$  ;
   pour chacun des successeurs S de R faire
2    $(\text{pref}, \text{post}) \leftarrow$ 
      $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$  ;
   fin
3  $R.\text{numPostfix} \leftarrow \text{post}$  ;  $\text{post}++$  ;
   retourner  $(\text{pref}, \text{post})$ 

```

R A pref 5 post 3

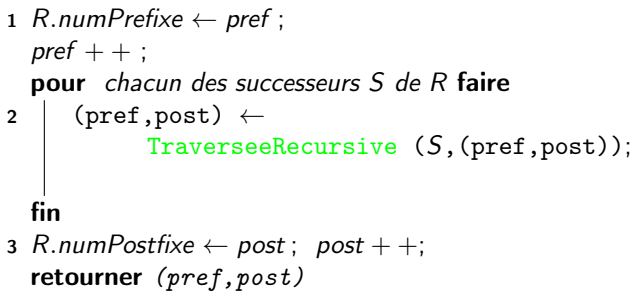
B

A

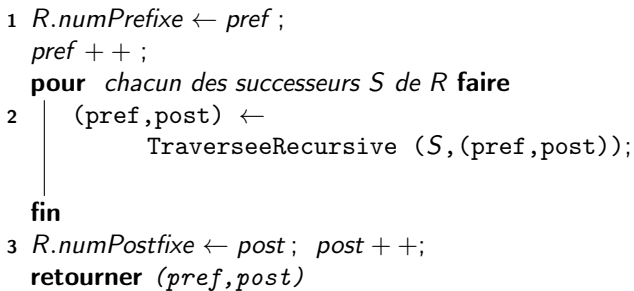
R

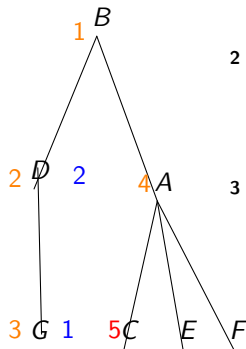
 pref

 post



B	
A	
E	F



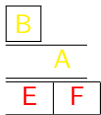


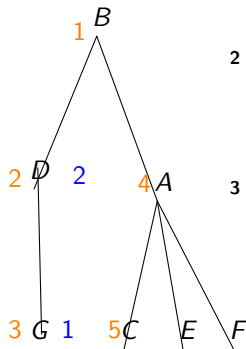
```

1  R.numPrefix  $\leftarrow$  pref ;
   pref ++ ;
   pour chacun des successeurs S de R faire
2  |   (pref,post)  $\leftarrow$ 
      |       TraverseeRecursive (S,(pref,post));
   |
   fin
3  R.numPostfixe  $\leftarrow$  post ; post ++ ;
   retourner (pref,post)

```

R C *pref* 5 *post* 3



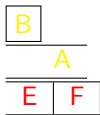


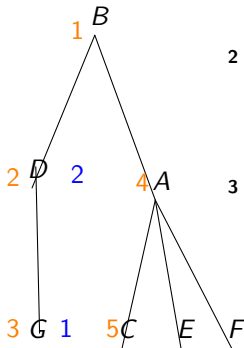
```

1 R.numPrefix ← pref ;
  pref ++;
  pour chacun des successeurs S de R faire
2   (pref,post) ←
    TraverseeRecursive (S,(pref,post));
  fin
3 R.numPostfix ← post ; post ++;
  retourner (pref,post)

```

R C pref 6 post 3



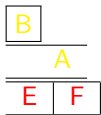


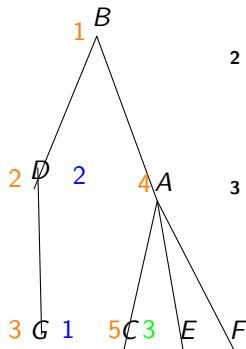
```

1  $R.\text{numPrefix} \leftarrow \text{pref}$  ;
    $\text{pref}++$  ;
   pour chacun des successeurs S de R faire
2    $(\text{pref}, \text{post}) \leftarrow$ 
      $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$  ;
   fin
3  $R.\text{numPostfix} \leftarrow \text{post}$  ;  $\text{post}++$  ;
   retourner  $(\text{pref}, \text{post})$ 

```

R C pref 6 post 3



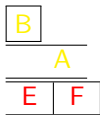


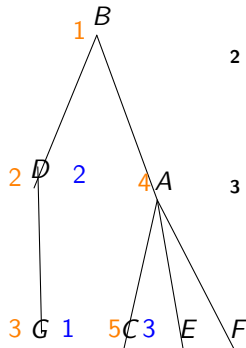
```

1  $R.numPrefix \leftarrow pref$  ;
   $pref++$  ;
  pour chacun des successeurs S de R faire
2    $(pref, post) \leftarrow$ 
      $TraverseeRecursive(S, (pref, post))$  ;
  fin
3  $R.numPostfixe \leftarrow post$  ;  $post++$  ;
  retourner  $(pref, post)$ 

```

R C pref 6 post 3



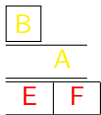


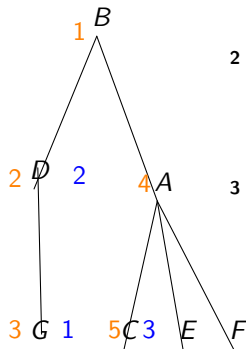
```

1  $R.numPrefix \leftarrow pref$  ;
    $pref ++$  ;
   pour chacun des successeurs S de R faire
2    $(pref, post) \leftarrow$ 
      $TraverseeRecursive (S, (pref, post))$ ;
   fin
3  $R.numPostfix \leftarrow post$  ;  $post ++$  ;
   retourner  $(pref, post)$ 

```

R C pref 6 post 4



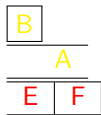


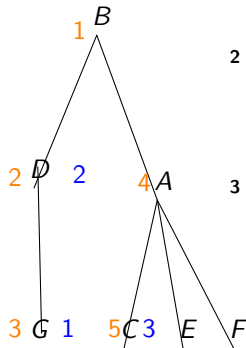
```

1  $R.\text{numPrefix} \leftarrow \text{pref}$  ;
   $\text{pref}++$  ;
  pour chacun des successeurs S de R faire
2    $(\text{pref}, \text{post}) \leftarrow$ 
      $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$  ;
  fin
3  $R.\text{numPostfix} \leftarrow \text{post}$  ;  $\text{post}++$  ;
  retourner  $(\text{pref}, \text{post})$ 

```

R C pref 6 post 4



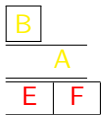


```

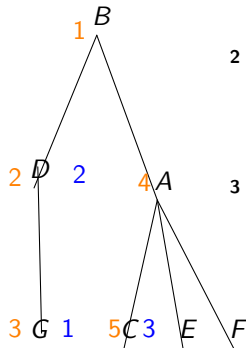
1 R.numPrefix ← pref ;
  pref ++ ;
  pour chacun des successeurs S de R faire
2   (pref,post) ←
    TraverseeRecursive (S,(pref,post));
  fin
3 R.numPostfixe ← post ; post ++ ;
  retourner (pref,post)

```

R C pref 6 post 4



On sort de l'appel TraverseeRecursive (C,5,3)

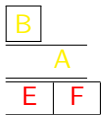


```

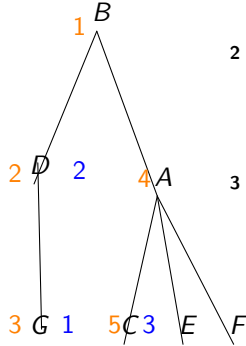
1  $R.\text{numPrefix} \leftarrow \text{pref}$  ;
    $\text{pref}++$  ;
   pour chacun des successeurs S de R faire
2      $(\text{pref}, \text{post}) \leftarrow$ 
        $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$  ;
   fin
3  $R.\text{numPostfix} \leftarrow \text{post}$  ;  $\text{post}++$  ;
   retourner  $(\text{pref}, \text{post})$ 

```

R C pref 6 post 4



On sort de l'appel $\text{TraverseeRecursive}(C, 5, 3)$



```

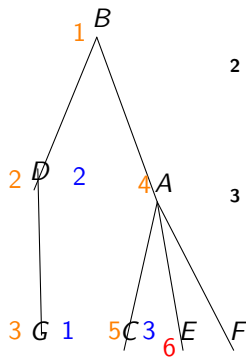
1 R.numPrefix ← pref ;
  pref ++ ;
  pour chacun des successeurs S de R faire
2   (pref,post) ←
    TraverseeRecursive (S,(pref,post));
  fin
3 R.numPostfixe ← post ; post ++ ;
  retourner (pref,post)

```

R E pref 6 post 4



On sort de l'appel TraverseeRecursive (C,5,3)
donc on continue la répétitive avec S=E

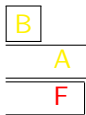


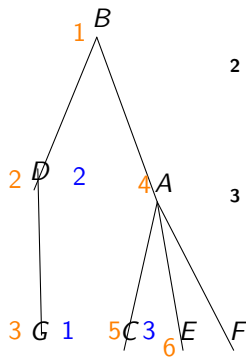
```

1  R.numPrefix  $\leftarrow$  pref ;
   pref ++ ;
   pour chacun des successeurs S de R faire
2  |   (pref,post)  $\leftarrow$ 
      |       TraverseeRecursive (S,(pref,post));
   |
   fin
3  R.numPostfixe  $\leftarrow$  post ; post ++ ;
   retourner (pref,post)

```

R E *pref* 6 *post* 4



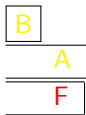


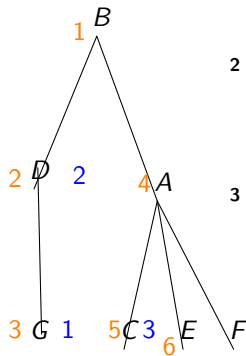
```

1  $R.\text{numPrefix} \leftarrow \text{pref}$  ;
    $\text{pref}++$  ;
   pour chacun des successeurs S de R faire
2      $(\text{pref}, \text{post}) \leftarrow$ 
        $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$  ;
   fin
3  $R.\text{numPostfix} \leftarrow \text{post}$  ;  $\text{post}++$  ;
   retourner  $(\text{pref}, \text{post})$ 

```

R E pref 7 post 4



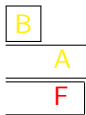


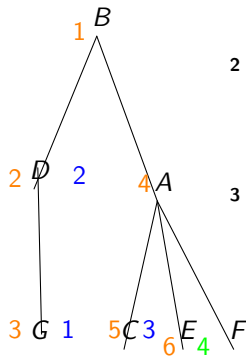
```

1  $R.\text{numPrefix} \leftarrow \text{pref}$  ;
    $\text{pref} ++$  ;
   pour chacun des successeurs S de R faire
2    $(\text{pref}, \text{post}) \leftarrow$ 
      $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$ ;
   fin
3  $R.\text{numPostfix} \leftarrow \text{post}$  ;  $\text{post} ++$  ;
   retourner  $(\text{pref}, \text{post})$ 

```

R E pref 7 post 4



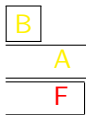


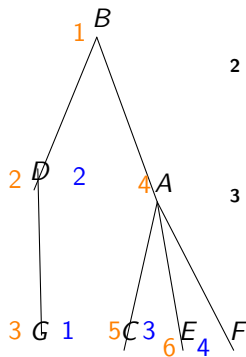
```

1  $R.numPrefix \leftarrow pref$  ;
    $pref ++$  ;
   pour chacun des successeurs S de R faire
2    $(pref, post) \leftarrow$ 
       $TraverseeRecursive (S, (pref, post))$ ;
   fin
3  $R.numPostfixe \leftarrow post$  ;  $post ++$  ;
   retourner  $(pref, post)$ 

```

R E pref 7 post 4



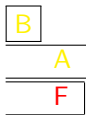


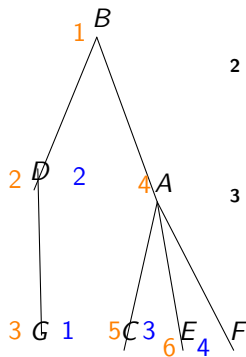
```

1  $R.\text{numPrefix} \leftarrow \text{pref}$  ;
    $\text{pref}++$  ;
   pour chacun des successeurs S de R faire
2      $(\text{pref}, \text{post}) \leftarrow$ 
        $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$  ;
   fin
3  $R.\text{numPostfix} \leftarrow \text{post}$  ;  $\text{post}++$  ;
   retourner  $(\text{pref}, \text{post})$ 

```

R E pref 7 post 4





```

1  $R.numPrefix \leftarrow pref$  ;  

   $pref ++$  ;  

  pour chacun des successeurs S de R faire  

2    $(pref, post) \leftarrow$   

     $TraverseeRecursive (S, (pref, post))$  ;  

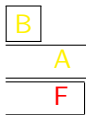
  fin  

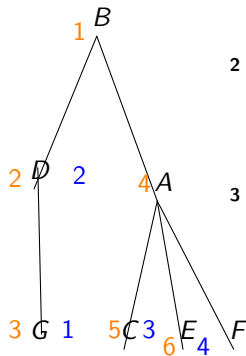
3  $R.numPostfix \leftarrow post$  ;  $post ++$  ;  

  retourner  $(pref, post)$ 

```

R E pref 7 post 5



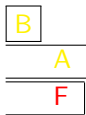


```

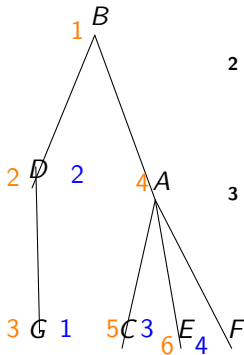
1   $R.\text{numPrefix} \leftarrow \text{pref}$  ;
    $\text{pref}++$  ;
   pour chacun des successeurs S de R faire
2  |    $(\text{pref}, \text{post}) \leftarrow$ 
      |    $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$ ;
   fin
3   $R.\text{numPostfix} \leftarrow \text{post}$  ;  $\text{post}++$  ;
   retourner  $(\text{pref}, \text{post})$ 

```

R E pref 7 post 5



On sort de l'appel $\text{TraverseeRecursive}(E, 6, 4)$



```

1  $R.\text{numPrefix} \leftarrow \text{pref}$  ;  

   $\text{pref}++$  ;  

  pour chacun des successeurs S de R faire  

2    $(\text{pref}, \text{post}) \leftarrow$   

     $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$  ;  

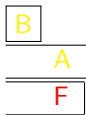
  fin  

3  $R.\text{numPostfix} \leftarrow \text{post}$  ;  $\text{post}++$  ;  

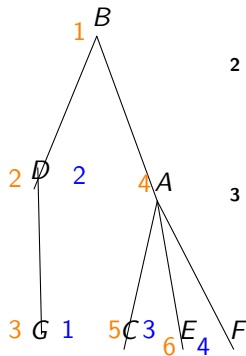
  retourner  $(\text{pref}, \text{post})$ 

```

R E pref 7 post 5



On sort de l'appel `TraverseeRecursive` (E,6,4)
 donc on continue la répétitive



```

1  R.numPrefix ← pref ;
   pref ++ ;
   pour chacun des successeurs S de R faire
2  |   (pref,post) ←
       TraverseeRecursive (S,(pref,post));
   fin
3  R.numPostfix ← post ; post ++ ;
   retourner (pref,post)

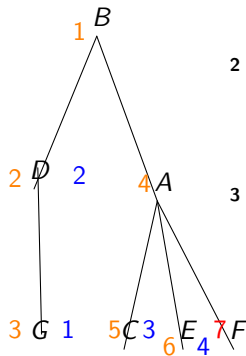
```

R F pref 7 post 5

B

A

On sort de l'appel TraverseeRecursive (E,6,4)
 donc on continue la répétitive avec S=F



```

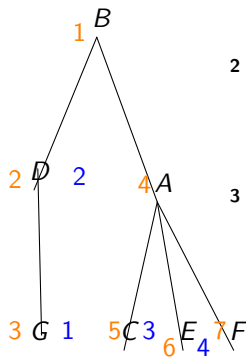
1  R.numPrefix  $\leftarrow$  pref ;
   pref ++ ;
   pour chacun des successeurs S de R faire
2      (pref,post)  $\leftarrow$ 
         TraverseeRecursive (S,(pref,post));
   fin
3  R.numPostfixe  $\leftarrow$  post ; post ++ ;
   retourner (pref,post)

```

R F *pref* 7 *post* 5

B

A



```

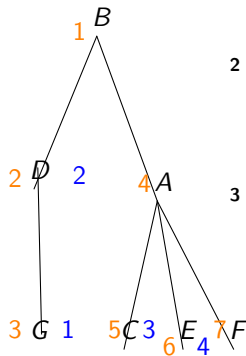
1  $R.\text{numPrefix} \leftarrow \text{pref}$  ;
    $\text{pref}++$  ;
   pour chacun des successeurs S de R faire
2    $(\text{pref}, \text{post}) \leftarrow$ 
      $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$  ;
   fin
3  $R.\text{numPostfix} \leftarrow \text{post}$  ;  $\text{post}++$  ;
   retourner  $(\text{pref}, \text{post})$ 

```

R F pref 8 post 5

B

A



```

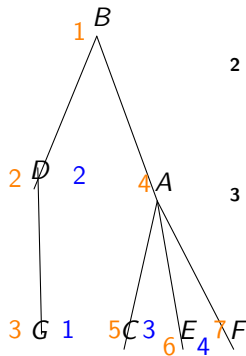
1  $R.numPrefix \leftarrow pref$  ;
    $pref ++$  ;
   pour chacun des successeurs S de R faire
2   |  $(pref, post) \leftarrow$ 
       $TraverseeRecursive (S, (pref, post))$ ;
   fin
3  $R.numPostfix \leftarrow post$  ;  $post ++$  ;
   retourner  $(pref, post)$ 

```

R F pref 8 post 5

B

A



```

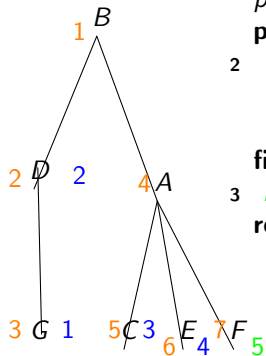
1  $R.\text{numPrefix} \leftarrow \text{pref}$  ;
    $\text{pref}++$  ;
   pour chacun des successeurs S de R faire
2    $(\text{pref}, \text{post}) \leftarrow$ 
      $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$  ;
   fin
3  $R.\text{numPostfix} \leftarrow \text{post}$  ;  $\text{post}++$  ;
   retourner  $(\text{pref}, \text{post})$ 

```

R F pref 8 post 5

B

A



```

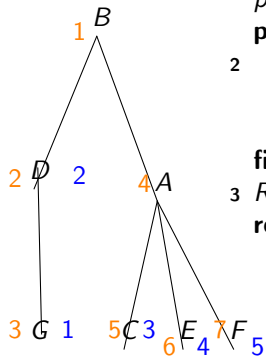
1  $R.numPrefixe \leftarrow pref$  ;
   $pref ++$  ;
  pour chacun des successeurs S de R faire
2    $(pref, post) \leftarrow$ 
      $TraverseeRecursive (S, (pref, post))$ ;
  fin
3  $R.numPostfixe \leftarrow post$  ;  $post ++$  ;
  retourner  $(pref, post)$ 

```

R F pref 8 post 5

B

A



```

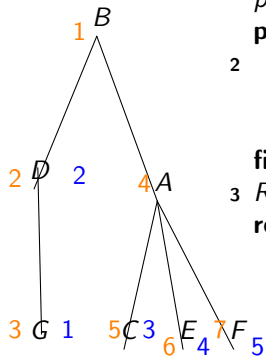
1  $R.\text{numPrefixe} \leftarrow \text{pref}$  ;
   $\text{pref}++$  ;
  pour chacun des successeurs S de R faire
2    $(\text{pref}, \text{post}) \leftarrow$ 
      $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$  ;
  fin
3  $R.\text{numPostfixe} \leftarrow \text{post}$  ;  $\text{post}++$  ;
  retourner  $(\text{pref}, \text{post})$ 

```

R F pref 8 post 6

B

A



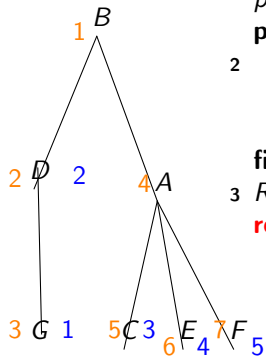
```

1  $R.\text{numPrefixe} \leftarrow \text{pref}$  ;
   $\text{pref}++$  ;
  pour chacun des successeurs S de R faire
2    $(\text{pref}, \text{post}) \leftarrow$ 
      $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$  ;
  fin
3  $R.\text{numPostfixe} \leftarrow \text{post}$  ;  $\text{post}++$  ;
  retourner  $(\text{pref}, \text{post})$ 

```

R F pref 8 post 6

B
A



```

1   $R.\text{numPrefix} \leftarrow \text{pref}$  ;
    $\text{pref}++$  ;
   pour chacun des successeurs S de R faire
2  |    $(\text{pref}, \text{post}) \leftarrow$ 
      |    $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$ ;
   fin
3   $R.\text{numPostfix} \leftarrow \text{post}$  ;  $\text{post}++$  ;
   retourner  $(\text{pref}, \text{post})$ 

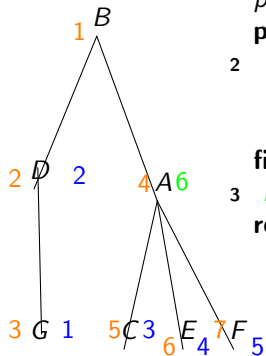
```

R F pref 8 post 6



On sort de l'appel $\text{TraverseeRecursive}$
donc de la répétitive

(F,7,5)



```

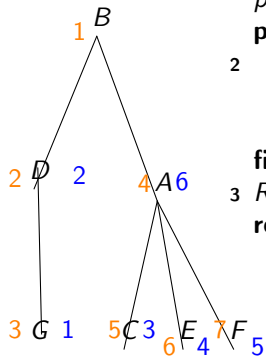
1  R.numPrefix ← pref ;
   pref ++ ;
   pour chacun des successeurs S de R faire
2  |   (pref,post) ←
       TraverseeRecursive (S,(pref,post));
   fin
3  R.numPostfixe ← post ; post ++ ;
   retourner (pref,post)

```

R A pref 8 post 6



On sort de l'appel TraverseeRecursive (F,7,5)
 donc de la répétitive et on sort donc de la répétitive de l'appel



```

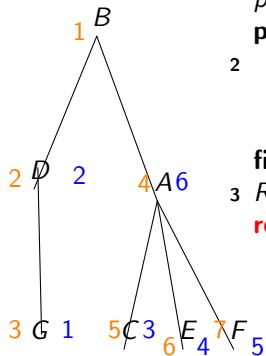
1  R.numPrefix ← pref ;
   pref ++ ;
   pour chacun des successeurs S de R faire
2  |   (pref,post) ←
      TraverseeRecursive (S,(pref,post));
   fin
3  R.numPostfixe ← post ;   post ++ ;
   retourner (pref,post)

```

R A pref 8 post 7

B

A



```

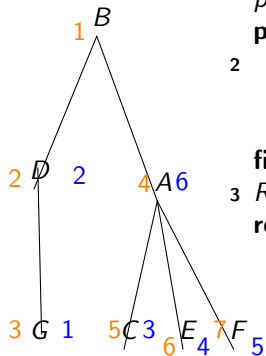
1 R.numPrefix ← pref ;
  pref ++ ;
  pour chacun des successeurs S de R faire
2   (pref,post) ←
     TraverseeRecursive (S,(pref,post));
  fin
3 R.numPostfix ← post ; post ++ ;
  retourner (pref,post)

```

R A pref 8 post 7



On sort de l'appel TraverseeRecursive (A,4,3)



```

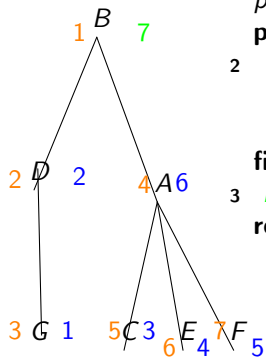
1  R.numPrefix ← pref ;
   pref ++ ;
   pour chacun des successeurs S de R faire
2  |   (pref,post) ←
       TraverseeRecursive (S,(pref,post));
   fin
3  R.numPostfix ← post ; post ++ ;
   retourner (pref,post)

```

R B pref 8 post 7



On sort de l'appel TraverseeRecursive (A,4,3)
 et on sort donc de la répétitive de l'appel TraverseeRecursive



```

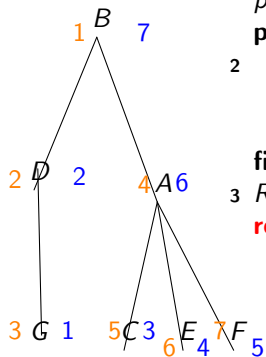
1  $R.\text{numPrefixe} \leftarrow \text{pref}$  ;
   $\text{pref}++$  ;
  pour chacun des successeurs S de R faire
2    $(\text{pref}, \text{post}) \leftarrow$ 
      $\text{TraverseeRecursive}(S, (\text{pref}, \text{post}))$  ;
  fin
3  $R.\text{numPostfixe} \leftarrow \text{post}$  ;  $\text{post}++$  ;
retourner  $(\text{pref}, \text{post})$ 

```

R B pref 8 post 7

B

A



```

1  R.numPrefix ← pref ;
   pref ++ ;
   pour chacun des successeurs S de R faire
2  |   (pref,post) ←
       TraverseeRecursive (S,(pref,post));
   |
   fin
3  R.numPostfix ← post ; post ++ ;
   retourner (pref,post)

```

R B pref 8 post 8

B

A

et on sort définitivement en renvoyant (8,8)