

Cours TP n°5

Ludovic Landuré

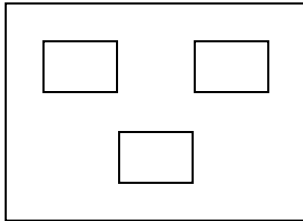
4 novembre 2008

Boucle for

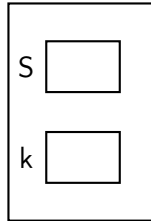
Boucle while

1^{er} exemple

```
[ > S :=0 :  
  for k from 1 to 5 do  
    S :=S+k ;  
  od :  
  S ;
```



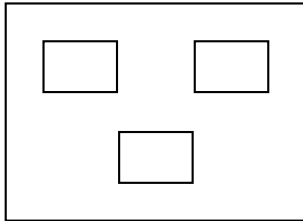
Processeur



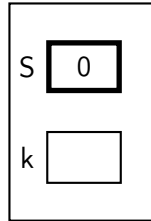
Mémoire

1^{er} exemple

```
[ > S :=0 :  
  for k from 1 to 5 do  
    S :=S+k ;  
  od :  
  S ;
```



Processeur



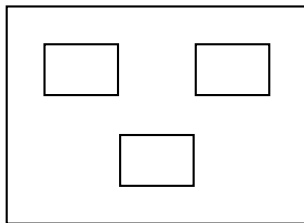
Mémoire

1^{er} exemple

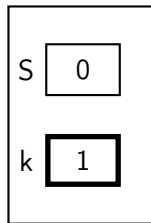
```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
  od :
  S ;

```



Processeur



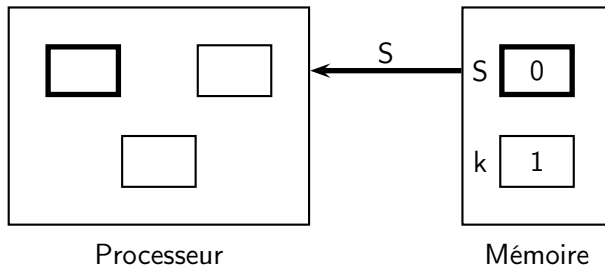
Mémoire

1^{er} exemple

```

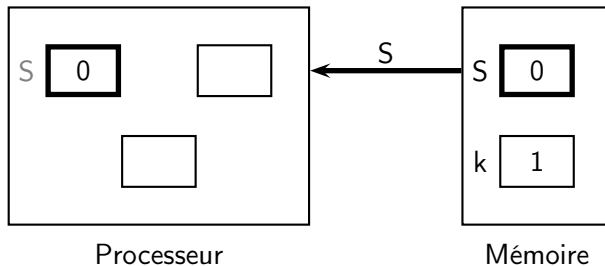
> S := 0 :
  for k from 1 to 5 do
    S := S+k ;
  od :
  S ;

```



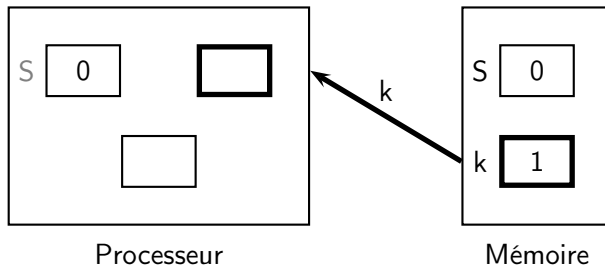
1^{er} exemple

```
[ > S :=0 :  
  for k from 1 to 5 do  
    S :=S+k ;  
  od :  
  S ;
```



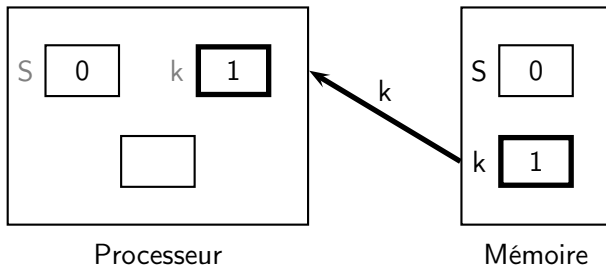
1^{er} exemple

```
> S := 0 :  
  for k from 1 to 5 do  
    S := S+k ;  
  od :  
  S ;
```



1^{er} exemple

```
[ > S :=0 :  
    for k from 1 to 5 do  
        S :=S+k ;  
    od :  
    S ;
```

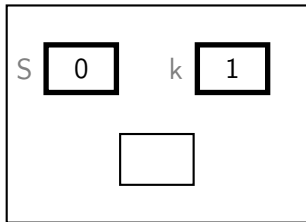


1^{er} exemple

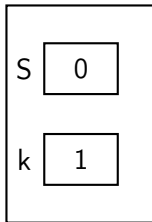
```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
  od :
  S ;

```



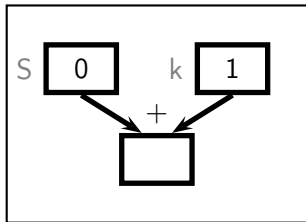
Processeur



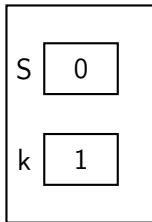
Mémoire

1^{er} exemple

```
[ > S :=0 :  
    for k from 1 to 5 do  
        S :=S+k ;  
    od :  
    S ;
```



Processeur



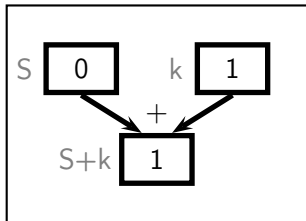
Mémoire

1^{er} exemple

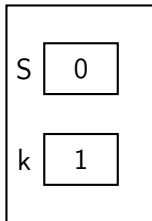
```

> S := 0 :
  for k from 1 to 5 do
    S := S+k ;
  od :
  S ;

```



Processeur



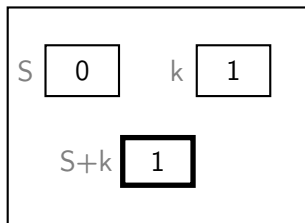
Mémoire

1^{er} exemple

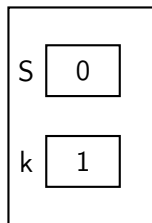
```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
  od :
  S ;

```



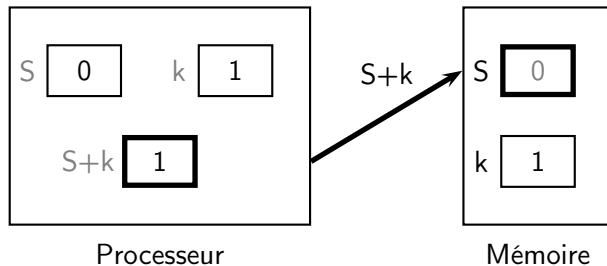
Processeur



Mémoire

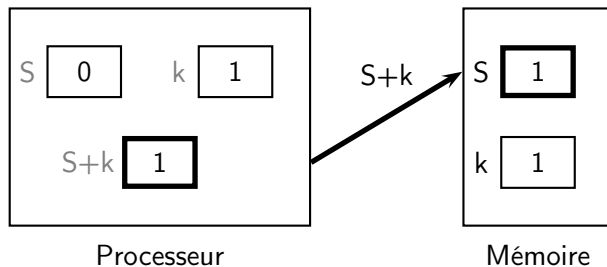
1^{er} exemple

```
[ > S :=0 :  
    for k from 1 to 5 do  
        S :=S+k ;  
    od :  
    S ;
```



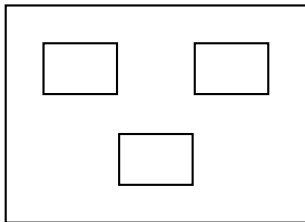
1^{er} exemple

```
[ > S :=0 :  
  for k from 1 to 5 do  
    S :=S+k ;  
  od :  
  S ;
```

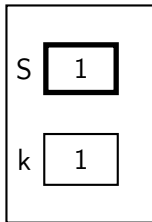


1^{er} exemple

```
[ > S :=0 :  
  for k from 1 to 5 do  
    S :=S+k ;  
  od :  
  S ;
```



Processeur



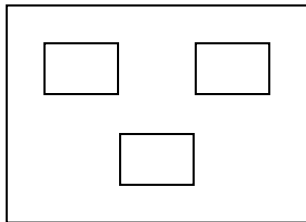
Mémoire

1^{er} exemple

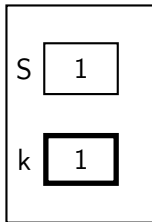
```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
    od :
    k :=k+1 ;
    S ;

```



Processeur



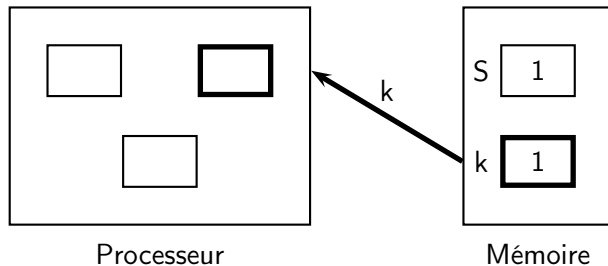
Mémoire

1^{er} exemple

```

[ > S :=0 :
    for k from 1 to 5 do
        S :=S+k;
        od :           k :=k+1;
    S;

```

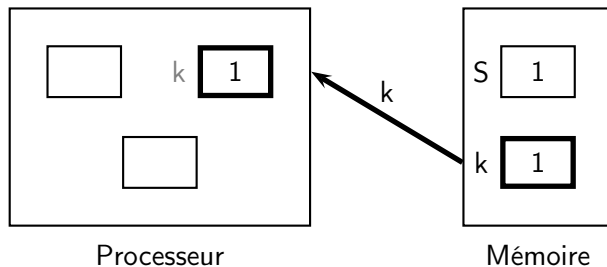


1^{er} exemple

```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
    od :      k :=k+1 ;
  S ;

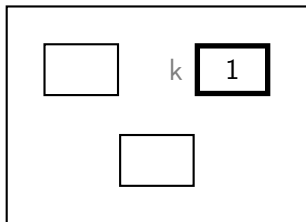
```



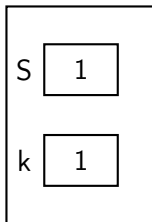
1^{er} exemple

```
> S :=0 :  
  for k from 1 to 5 do  
    S :=S+k ;  
  od :  
  S ;
```

$k := k+1 ;$



Processeur

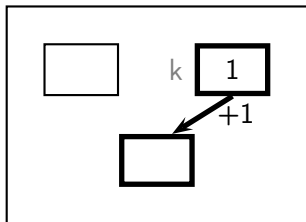


Mémoire

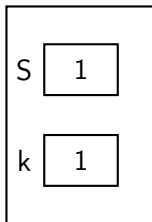
1^{er} exemple

```
> S :=0 :  
  for k from 1 to 5 do  
    S :=S+k ;  
  od :  
  S ;
```

$k := k+1 ;$



Processeur

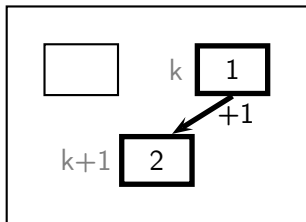


Mémoire

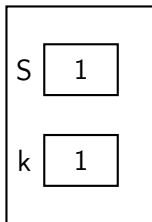
1^{er} exemple

```
> S :=0 :  
  for k from 1 to 5 do  
    S :=S+k ;  
  od :  
  S ;
```

$k := k+1 ;$



Processeur



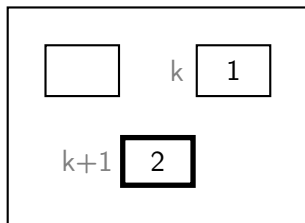
Mémoire

1^{er} exemple

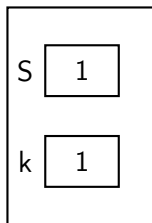
```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
    od :
    S ;
    k :=k+1 ;

```



Processeur

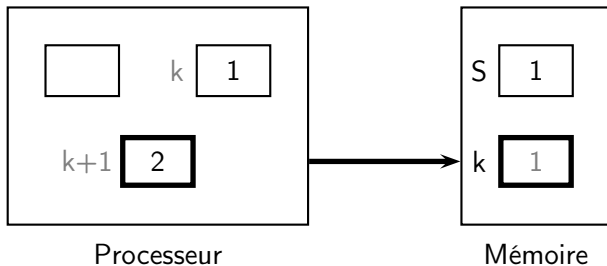


Mémoire

1^{er} exemple

```
> S :=0 :  
  for k from 1 to 5 do  
    S :=S+k ;  
  od :  
  S ;
```

$k := k + 1 ;$



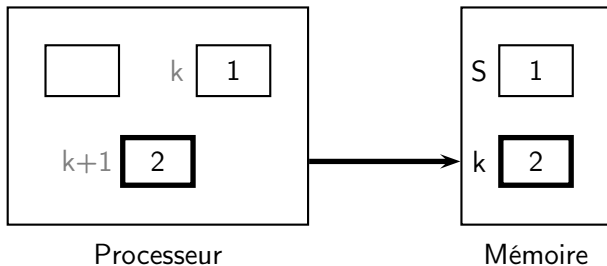
1^{er} exemple

```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
    od :
    S ;

```

$k := k + 1 ;$

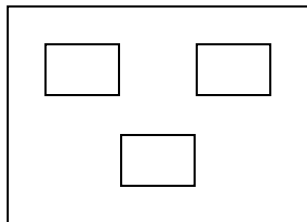


1^{er} exemple

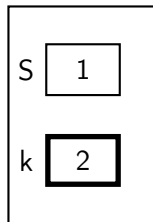
```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
    od :
    S ;
    k :=k+1 ;

```



Processeur



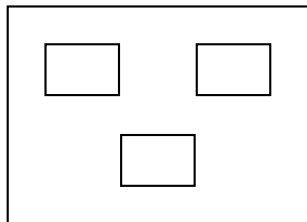
Mémoire

1^{er} exemple

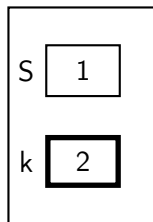
```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
  od :
  S ;

```



Processeur



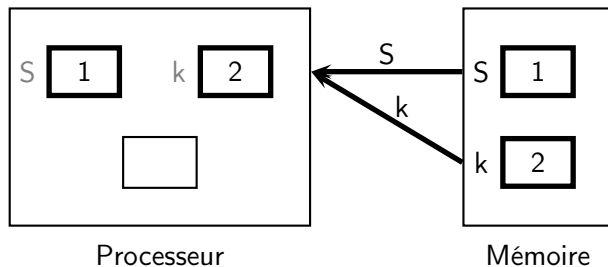
Mémoire

1^{er} exemple

```

[ > S :=0 :
    for k from 1 to 5 do
      S :=S+k ;
    od :
    S ;

```

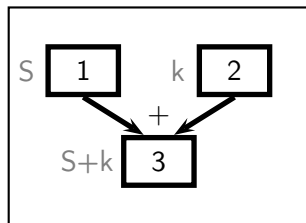


1^{er} exemple

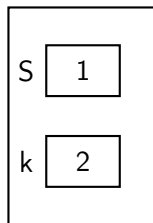
```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
  od :
  S ;

```



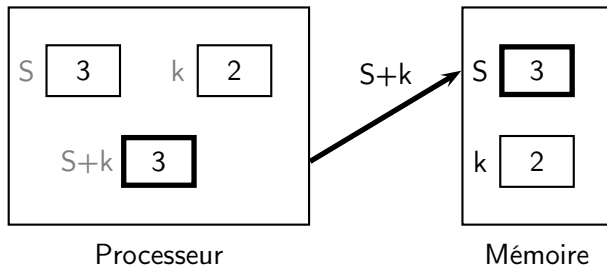
Processeur



Mémoire

1^{er} exemple

```
> S := 0 :  
  for k from 1 to 5 do  
    S := S+k ;  
  od :  
  S ;
```

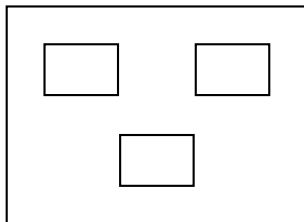


1^{er} exemple

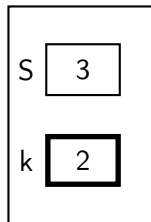
```

[ > S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
    od :
    S ;
    k :=k+1 ;

```



Processeur



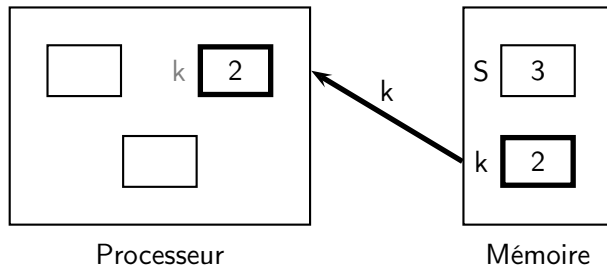
Mémoire

1^{er} exemple

```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
    od :      k :=k+1 ;
  S ;

```

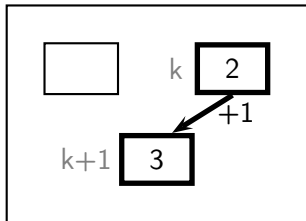


1^{er} exemple

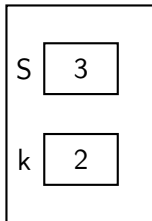
```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
    od :
    S ;
    k :=k+1 ;

```



Processeur



Mémoire

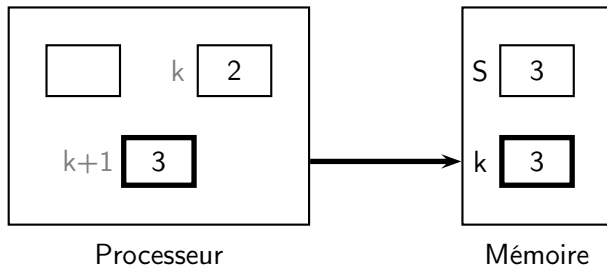
1^{er} exemple

```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
    od :
    S ;

```

$k := k+1 ;$

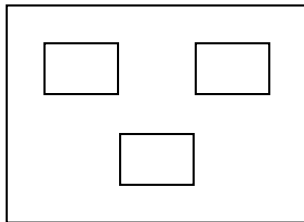


1^{er} exemple

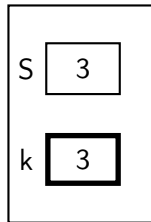
```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
  od :
  S ;

```



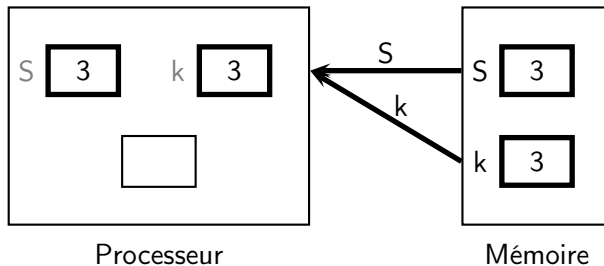
Processeur



Mémoire

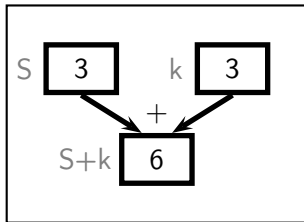
1^{er} exemple

```
[ > S :=0 :  
  for k from 1 to 5 do  
    S :=S+k ;  
  od :  
  S ;
```

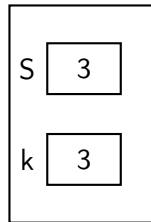


1^{er} exemple

```
> S := 0 :  
  for k from 1 to 5 do  
    S := S+k ;  
  od :  
  S ;
```



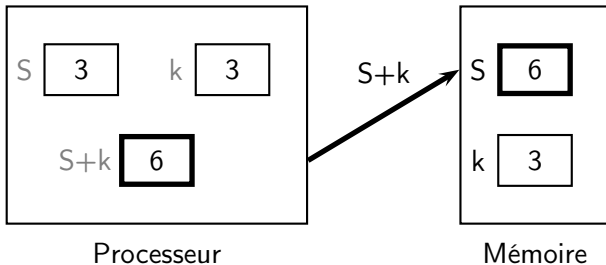
Processeur



Mémoire

1^{er} exemple

```
> S :=0 :  
  for k from 1 to 5 do  
    S :=S+k ;  
  od :  
  S ;
```

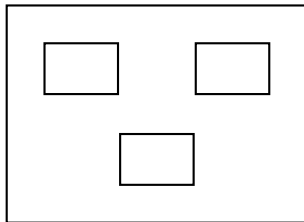


1^{er} exemple

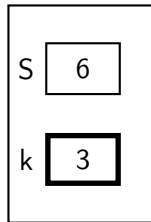
```

[ > S :=0 :
    for k from 1 to 5 do
        S :=S+k ;
        od :           k :=k+1 ;
        S ;

```



Processeur



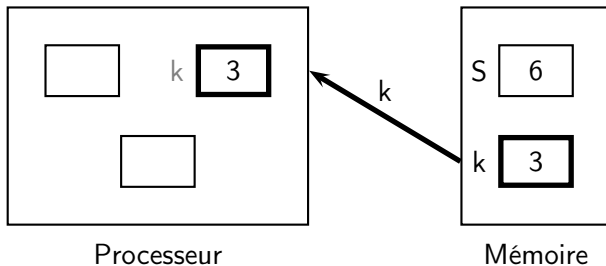
Mémoire

1^{er} exemple

```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
    od :      k :=k+1 ;
  S ;

```

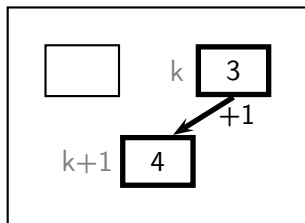


1^{er} exemple

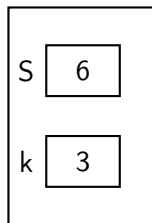
```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
    od :
    S ;
    k :=k+1 ;

```



Processeur



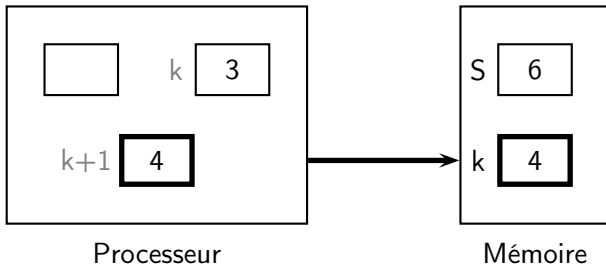
Mémoire

1^{er} exemple

```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
    od :
    S ;
    k :=k+1 ;

```

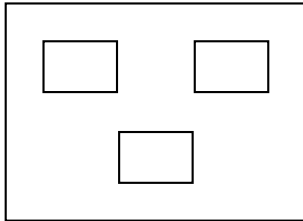


1^{er} exemple

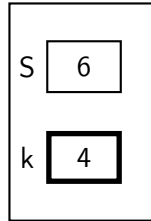
```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
  od :
  S ;

```



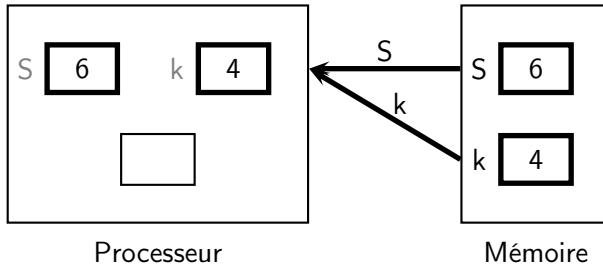
Processeur



Mémoire

1^{er} exemple

```
[ > S :=0 :  
  for k from 1 to 5 do  
    S :=S+k ;  
  od :  
  S ;
```

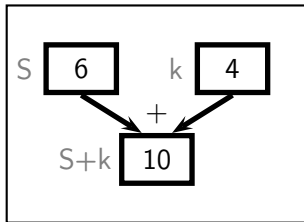


1^{er} exemple

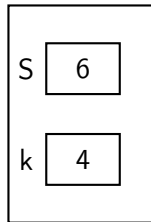
```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
  od :
  S ;

```



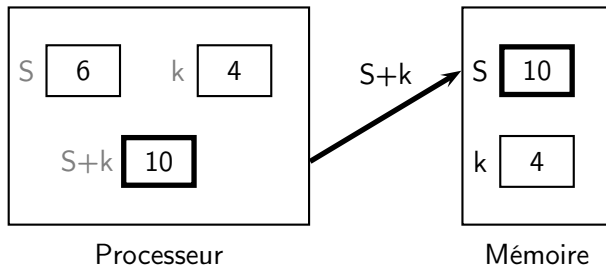
Processeur



Mémoire

1^{er} exemple

```
[ > S :=0 :  
  for k from 1 to 5 do  
    S :=S+k ;  
  od :  
  S ;
```

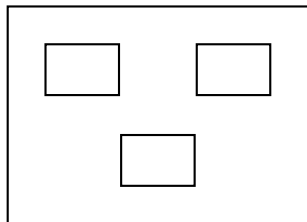


1^{er} exemple

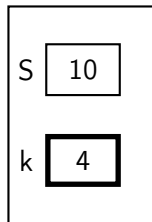
```

[ > S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
    od :
    S ;
    k :=k+1 ;

```



Processeur



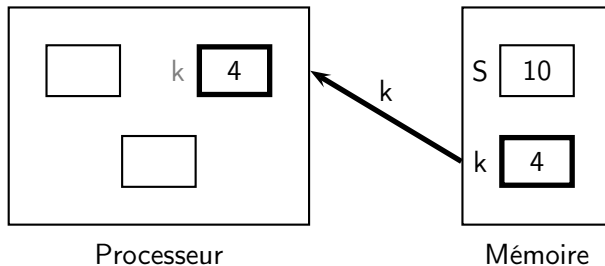
Mémoire

1^{er} exemple

```

[ > S :=0 :
    for k from 1 to 5 do
        S :=S+k ;
        od :           k :=k+1 ;
        S ;

```

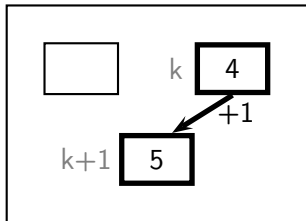


1^{er} exemple

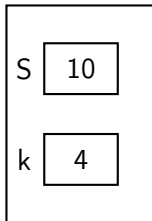
```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
    od :
    S ;
    k :=k+1 ;

```



Processeur



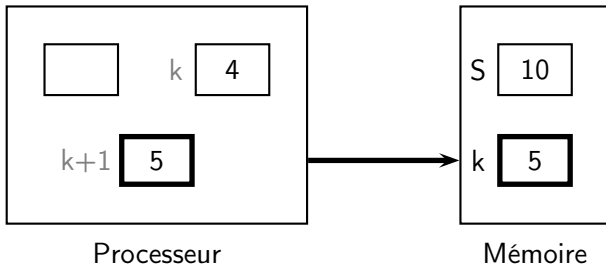
Mémoire

1^{er} exemple

```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
    od :
    S ;
    k :=k+1 ;

```

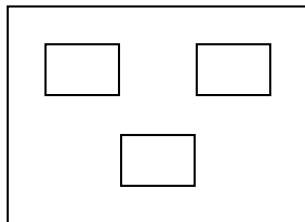


1^{er} exemple

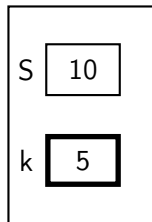
```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
  od :
  S ;

```



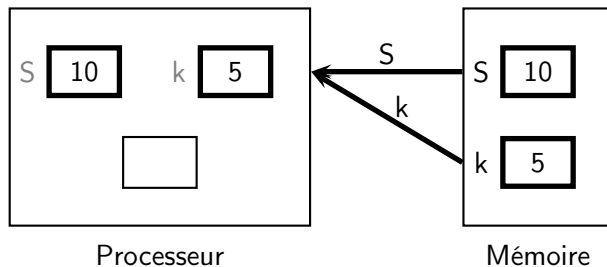
Processeur



Mémoire

1^{er} exemple

```
[ > S :=0 :  
    for k from 1 to 5 do  
        S :=S+k ;  
    od :  
    S ;
```

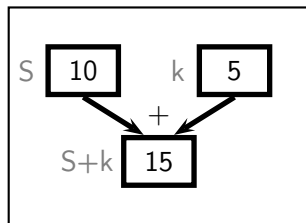


1^{er} exemple

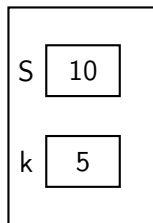
```

> S := 0 :
  for k from 1 to 5 do
    S := S+k ;
  od :
  S ;

```



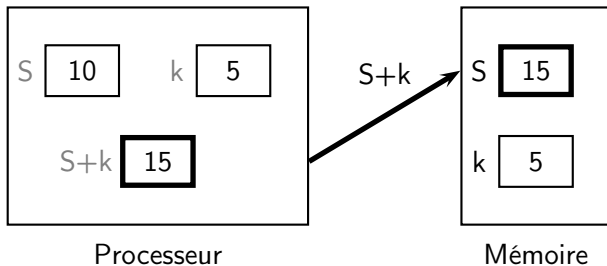
Processeur



Mémoire

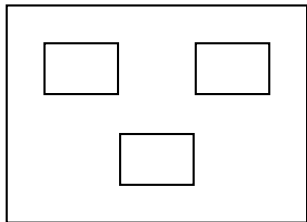
1^{er} exemple

```
> S := 0 :  
  for k from 1 to 5 do  
    S := S + k ;  
  od :  
  S ;
```

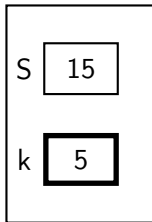


1^{er} exemple

```
[ > S :=0 :  
  for k from 1 to 5 do  
    S :=S+k ;  
  od :  
  S ;  
  k :=k+1 ;
```



Processeur



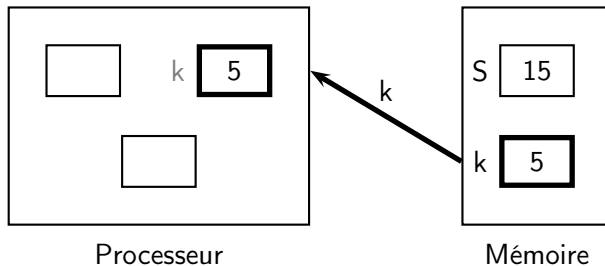
Mémoire

1^{er} exemple

```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
    od :      k :=k+1 ;
  S ;

```

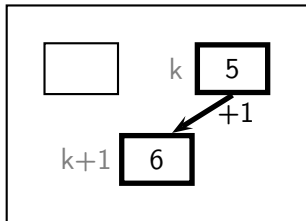


1^{er} exemple

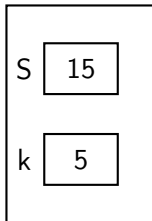
```

> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
    od :
    S ;
    k :=k+1 ;

```



Processeur



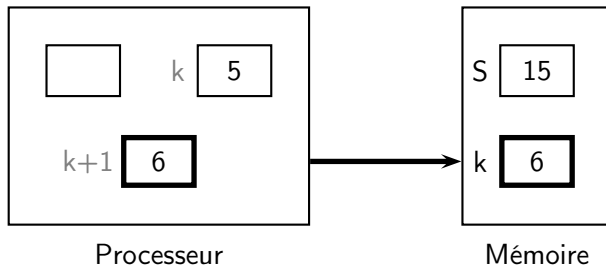
Mémoire

1^{er} exemple

```

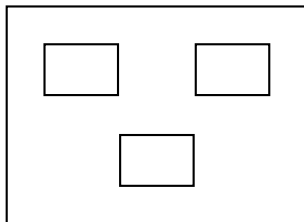
> S :=0 :
  for k from 1 to 5 do
    S :=S+k ;
    od :
    S ;
    k :=k+1 ;

```

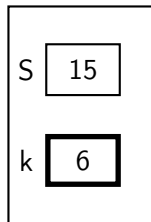


1^{er} exemple

```
[ > S :=0 :  
  for k from 1 to 5 do  
    S :=S+k ;  
  od :  
  S ;
```



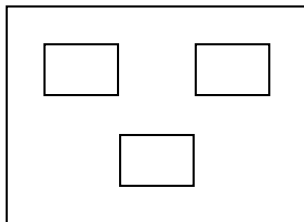
Processeur



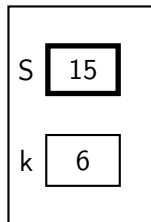
Mémoire

1^{er} exemple

```
[ > S :=0 :  
    for k from 1 to 5 do  
        S :=S+k ;  
    od :  
    S ;
```



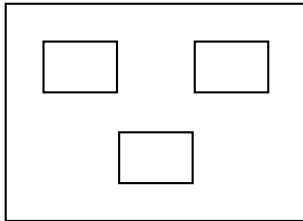
Processeur



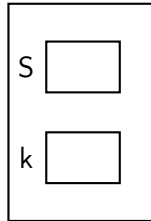
Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```



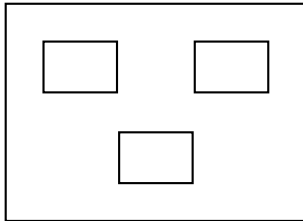
Processeur



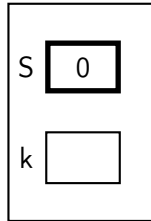
Mémoire

1^{er} exemple

```
> S :=0 :
  k :=0 :
  while S<7 do
    k :=k+1 ;
    S :=S+k ;
  od :
  k ;
```



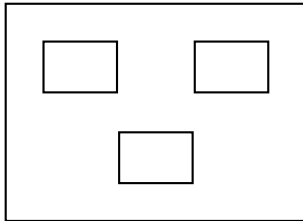
Processeur



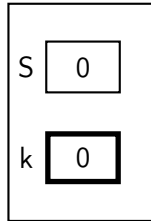
Mémoire

1^{er} exemple

```
> S :=0 :
  k :=0 :
  while S<7 do
    k :=k+1 ;
    S :=S+k ;
  od :
  k ;
```



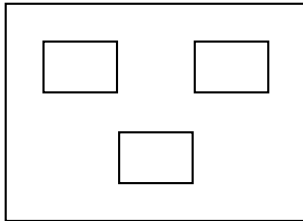
Processeur



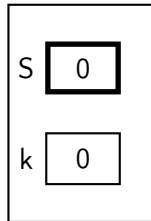
Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```



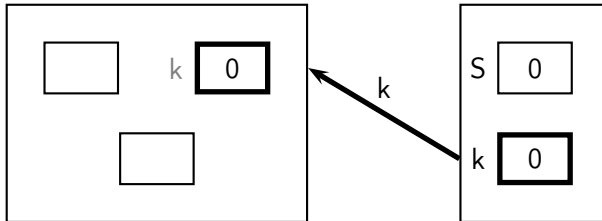
Processeur



Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```

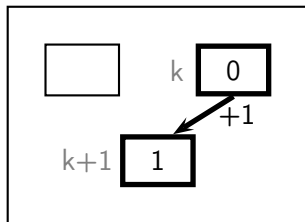


Processeur

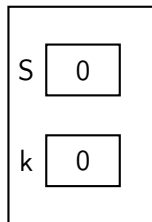
Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```



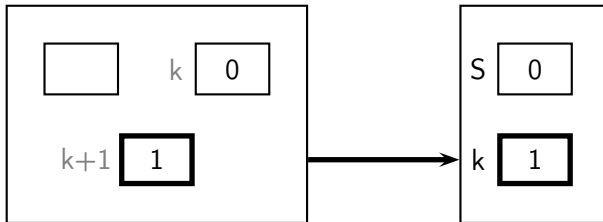
Processeur



Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```

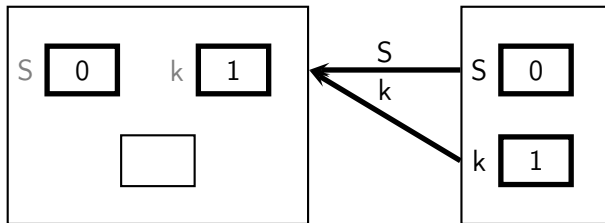


Processeur

Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```



Processeur

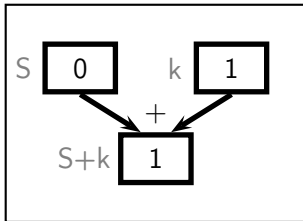
Mémoire

1^{er} exemple

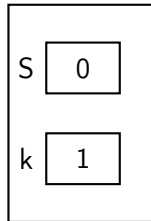
```

> S :=0 :
  k :=0 :
  while S<7 do
    k :=k+1 ;
    S :=S+k ;
  od :
  k ;

```



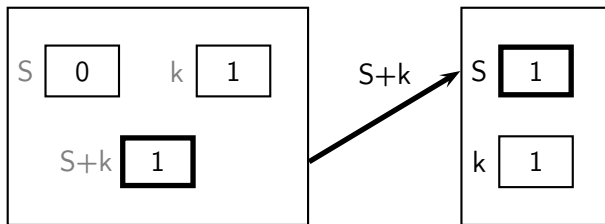
Processeur



Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```

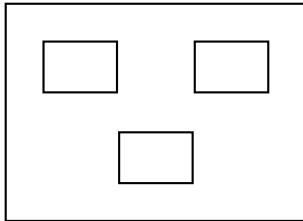


Processeur

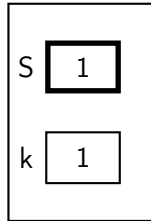
Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```



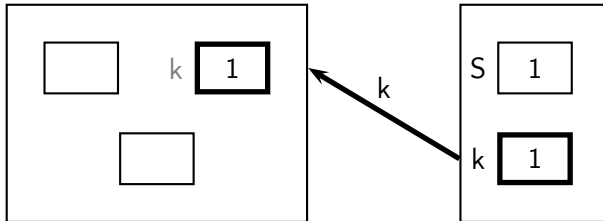
Processeur



Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```

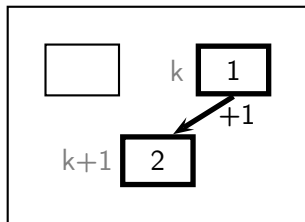


Processeur

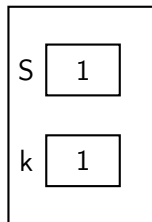
Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```



Processeur



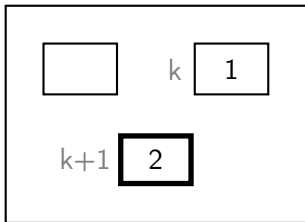
Mémoire

1^{er} exemple

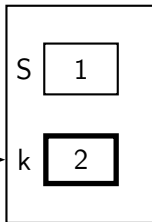
```

> S :=0 :
  k :=0 :
  while S<7 do
    k :=k+1 ;
    S :=S+k ;
  od :
  k ;

```



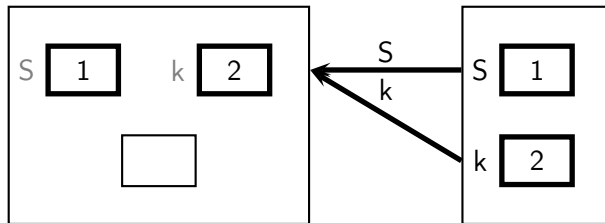
Processeur



Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```



Processeur

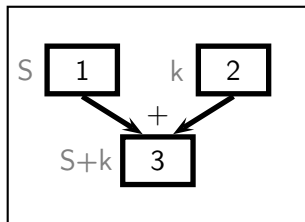
Mémoire

1^{er} exemple

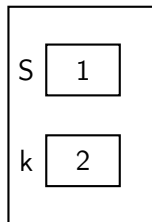
```

> S :=0 :
  k :=0 :
  while S<7 do
    k :=k+1 ;
    S :=S+k ;
  od :
  k ;

```



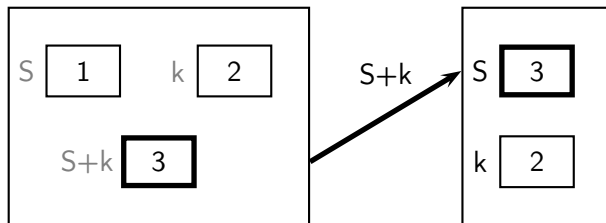
Processeur



Mémoire

1^{er} exemple

```
[ > S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```

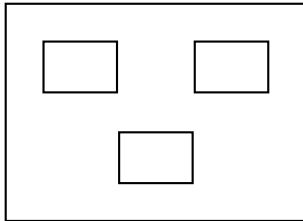


Processeur

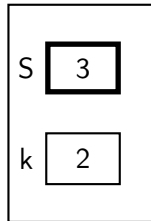
Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```



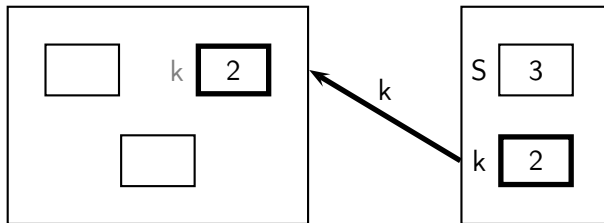
Processeur



Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```

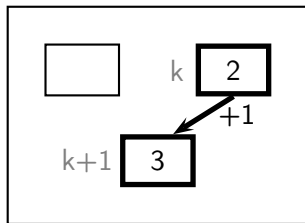


Processeur

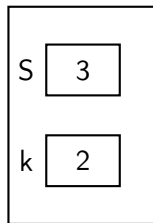
Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```



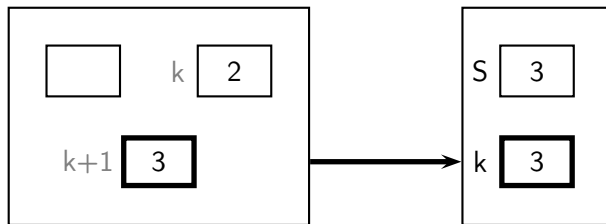
Processeur



Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```

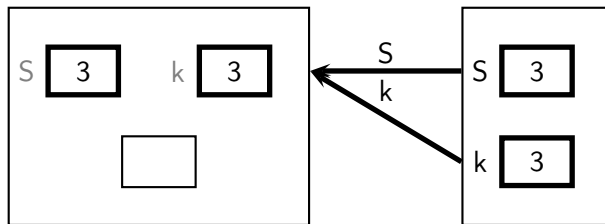


Processeur

Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```

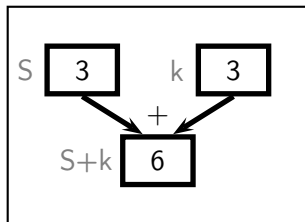


Processeur

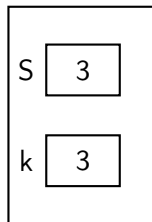
Mémoire

1^{er} exemple

```
[ > S :=0 :
    k :=0 :
    while S<7 do
    k :=k+1 ;
    S :=S+k ;
    od :
    k ;
```



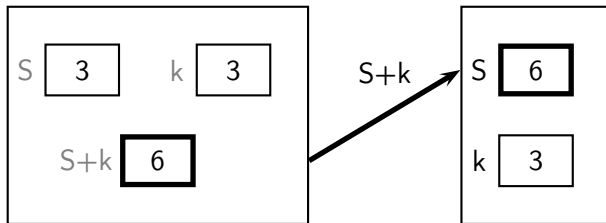
Processeur



Mémoire

1^{er} exemple

```
[ > S :=0 :
    k :=0 :
    while S<7 do
    k :=k+1 ;
    S :=S+k ;
    od :
    k ;
```

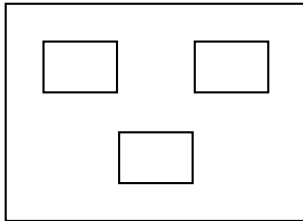


Processeur

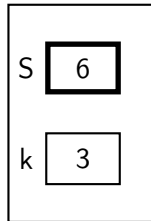
Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```



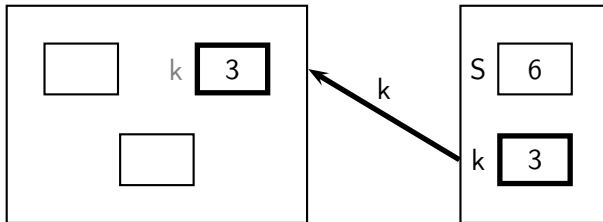
Processeur



Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```

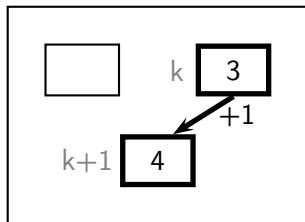


Processeur

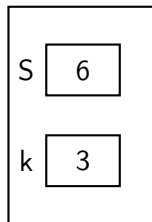
Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```



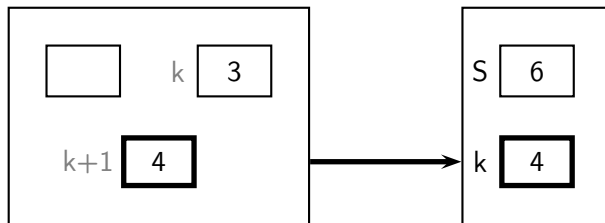
Processeur



Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```

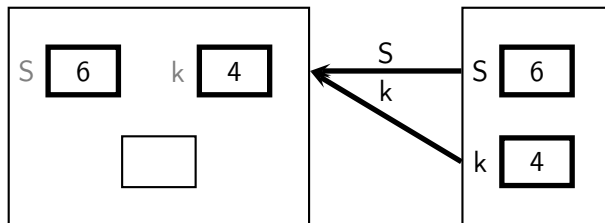


Processeur

Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```



Processeur

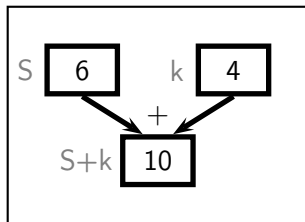
Mémoire

1^{er} exemple

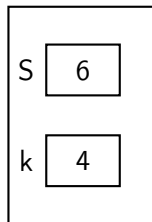
```

> S :=0 :
  k :=0 :
  while S<7 do
    k :=k+1 ;
    S :=S+k ;
  od :
  k ;

```



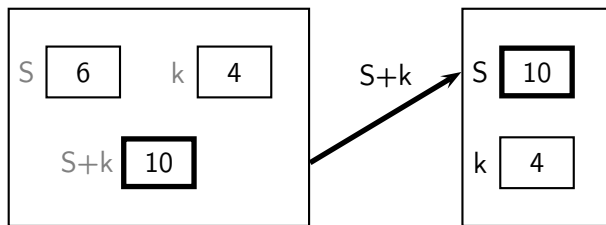
Processeur



Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```

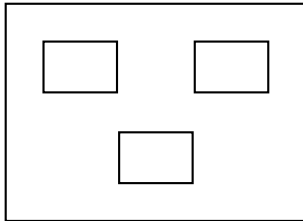


Processeur

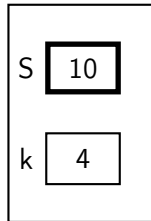
Mémoire

1^{er} exemple

```
> S :=0 :  
  k :=0 :  
  while S<7 do  
    k :=k+1 ;  
    S :=S+k ;  
  od :  
  k ;
```



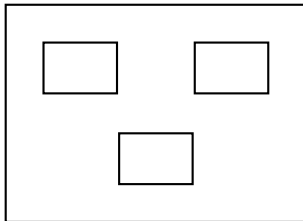
Processeur



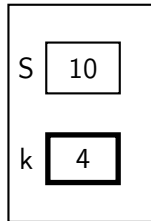
Mémoire

1^{er} exemple

```
> S :=0 :
  k :=0 :
  while S<7 do
    k :=k+1 ;
    S :=S+k ;
  od :
  k ;
```



Processeur



Mémoire