

Rechercher un élément dans une liste

- Comment faire ?

Pour chaque élément de la liste

Si c'est l'élément cherché, on a fini, retourner l'élément

Sinon, on continue

Si à la fin on n'a pas trouvé, c'est qu'il n'y est pas

- Quel est la signature de la fonction ?

Plusieurs possibilités:

- chercher un élément selon sa valeur ou son adresse
- Retourner un booléen ou un pointeur vers l'élément

Aujourd'hui

```
int_list rechercher(int_list liste, int val);
```

Rechercher un élément dans une liste

- Comment faire ?

Pour chaque élément de la liste

Si c'est l'élément cherché, on a fini, retourner l'élément

Sinon, on continue

Si à la fin on n'a pas trouvé, c'est qu'il n'y est pas

```
int_list rechercher(int_list liste, int val){  
    while(liste !=NULL){  
        if(liste->car==val)  
            return liste;  
        else  
            liste=liste->cdr;  
    }  
    return NULL,  
}
```

ATTENTION AU ELSE

Rechercher un élément dans une liste

- Comment faire ?

Pour chaque élément de la liste

Si c'est l'élément cherché, on a fini, retourner l'élément

Sinon, on continue

Si à la fin on n'a pas trouvé, c'est qu'il n'y est pas

```
int_list rechercher(int_list liste, int val){  
    while(liste !=NULL){  
        if(liste->car==val)  
            return liste;  
        liste=liste->cdr;  
    }  
    return NULL,  
}
```

On peut aussi ne pas le mettre

Rechercher un élément dans une liste

```
int_list rechercher(int_list liste, int val){
    while(liste !=NULL){
        if(liste->car==val)
            return liste;
        else
            liste=liste->cdr;
    }
    return NULL,
}
```

```
int main(){
    int_list ns, elt;
    ns=cons(1,cons(2,cons(3,NULL)));
    elt= rechercher(ns,2);
    if(elt!=NULL)
        printf("2 est présent\n »);
    return 0;
}
```

PILE

TAS

elt

ns

0x190

0x180

3

NULL

0x190

1

0x210

0x210

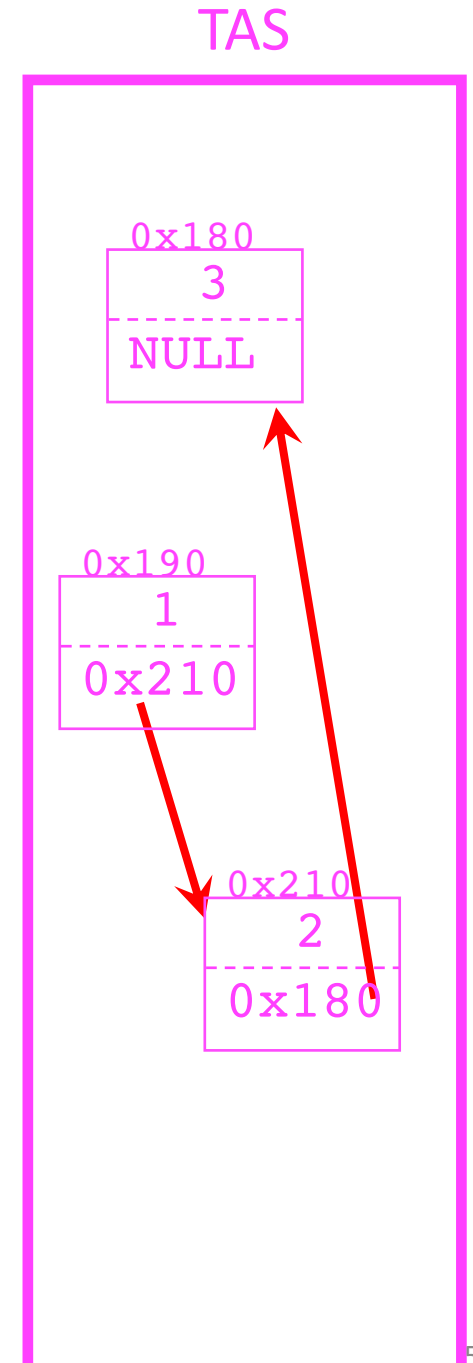
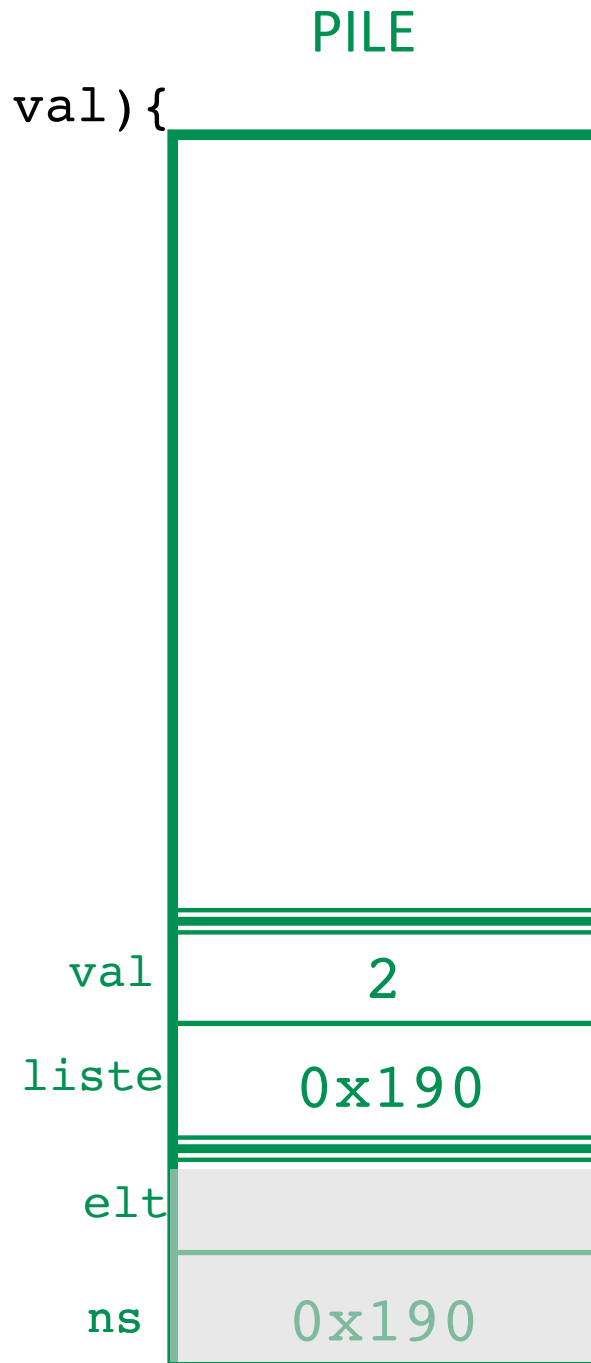
2

0x180

Rechercher un élément dans une liste

```
int_list rechercher(int_list liste, int val){  
→   while(liste !=NULL){  
       if(liste->car==val)  
           return liste;  
       else  
           liste=liste->cdr;  
   }  
   return NULL,  
}
```

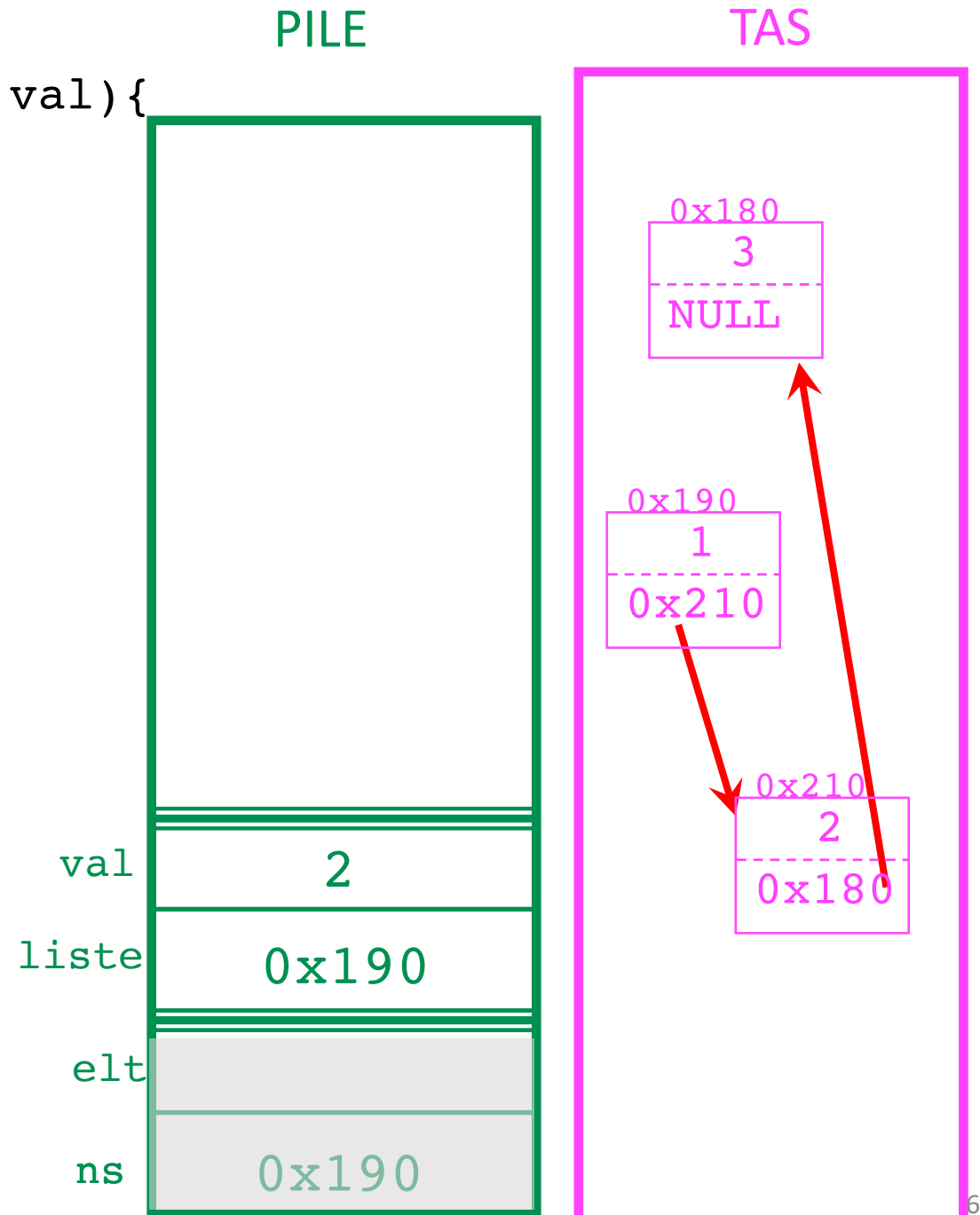
```
int main(){  
    int_list ns, elt;  
    ns=cons(1,cons(2,cons(3,NULL)));  
    elt= rechercher(ns,2);  
    if(elt!=NULL)  
        printf("2 est présent\n »);  
    return 0;  
}
```



Rechercher un élément dans une liste

```
int_list rechercher(int_list liste, int val){
    while(liste !=NULL){
        if(liste->car==val)
            return liste;
        else
            liste=liste->cdr;
    }
    return NULL,
}
```

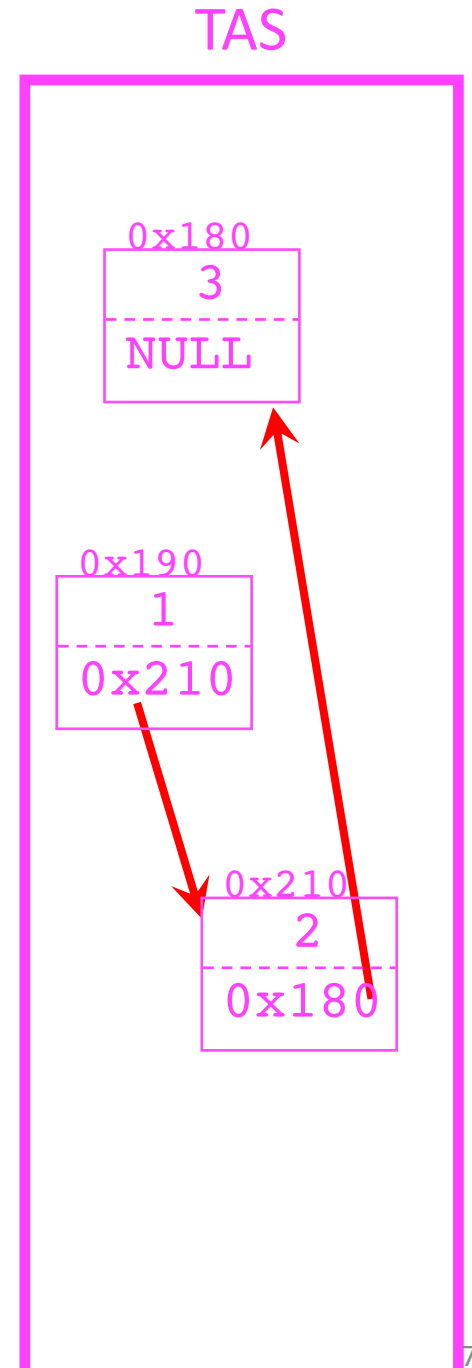
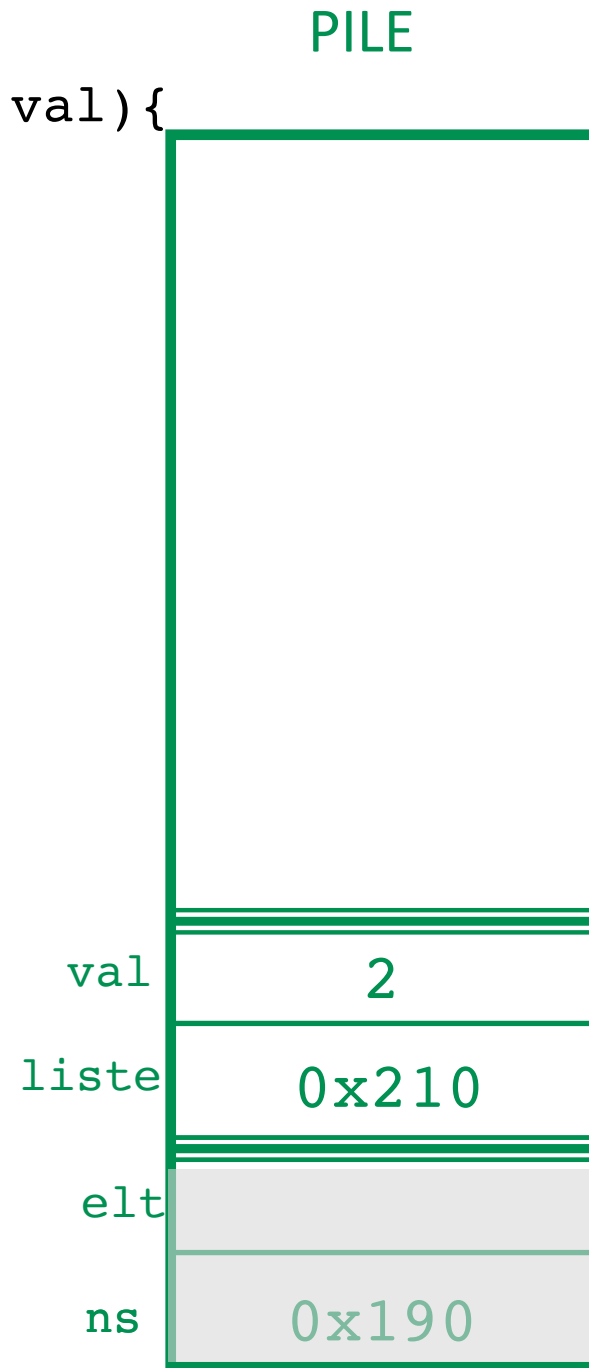
```
int main(){
    int_list ns, elt;
    ns=cons(1,cons(2,cons(3,NULL)));
    elt= rechercher(ns,2);
    if(elt!=NULL)
        printf("2 est présent\n »);
    return 0;
}
```



Rechercher un élément dans une liste

```
int_list rechercher(int_list liste, int val){
    while(liste !=NULL){
        if(liste->car==val)
            return liste;
        else
            liste=liste->cdr;
    }
    return NULL,
}
```

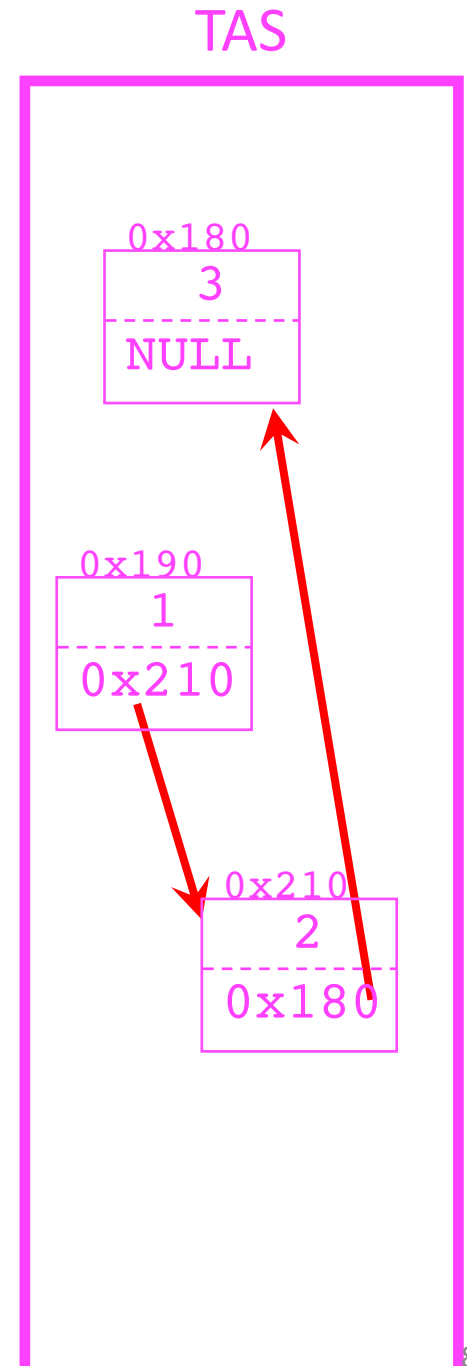
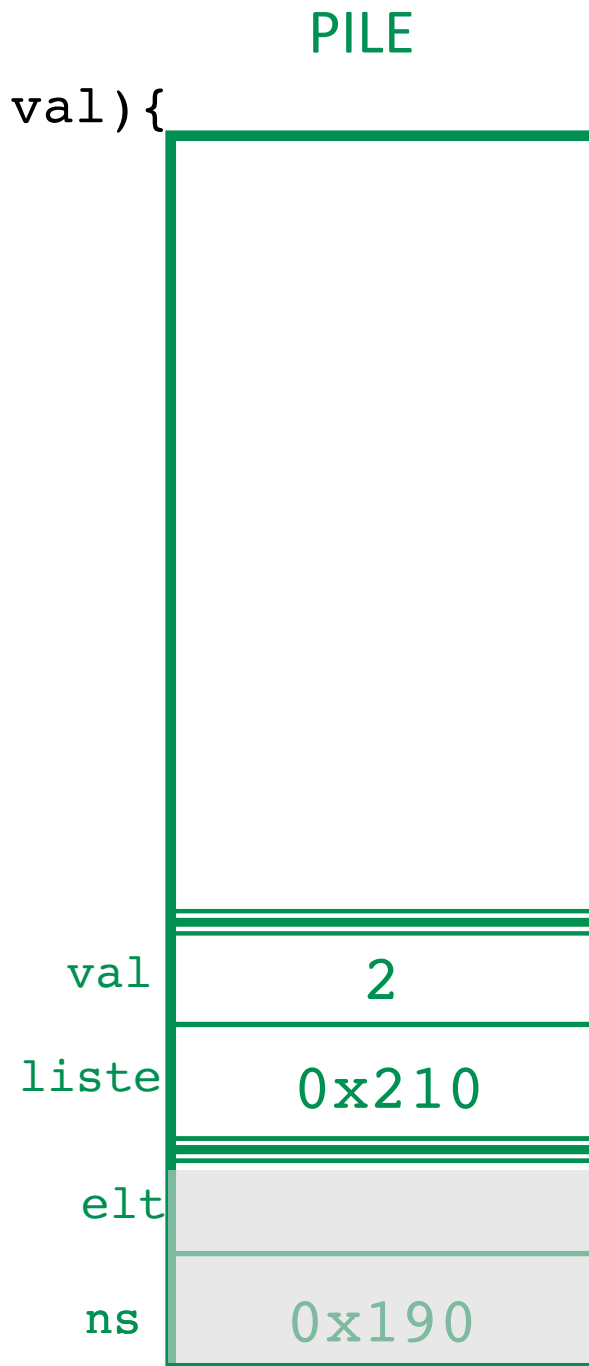
```
int main(){
    int_list ns, elt;
    ns=cons(1,cons(2,cons(3,NULL)));
    elt= rechercher(ns,2);
    if(elt!=NULL)
        printf("2 est présent\n »);
    return 0;
}
```



Rechercher un élément dans une liste

```
int_list rechercher(int_list liste, int val){  
→   while(liste !=NULL){  
       if(liste->car==val)  
           return liste;  
       else  
           liste=liste->cdr;  
   }  
   return NULL,  
}
```

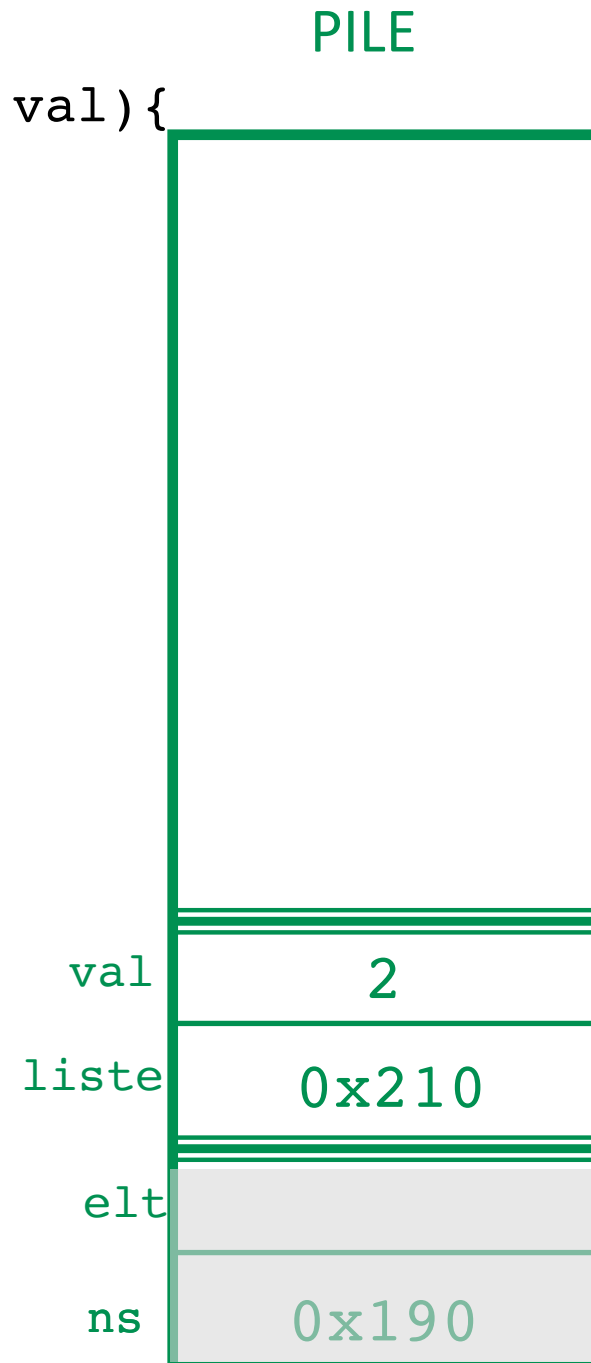
```
int main(){  
    int_list ns, elt;  
    ns=cons(1,cons(2,cons(3,NULL)));  
    elt= rechercher(ns,2);  
    if(elt!=NULL)  
        printf("2 est présent\n »);  
    return 0;  
}
```



Rechercher un élément dans une liste

```
int_list rechercher(int_list liste, int val){  
    while(liste !=NULL){  
        if(liste->car==val)  
            return liste;  
        else  
            liste=liste->cdr;  
    }  
    return NULL,  
}
```

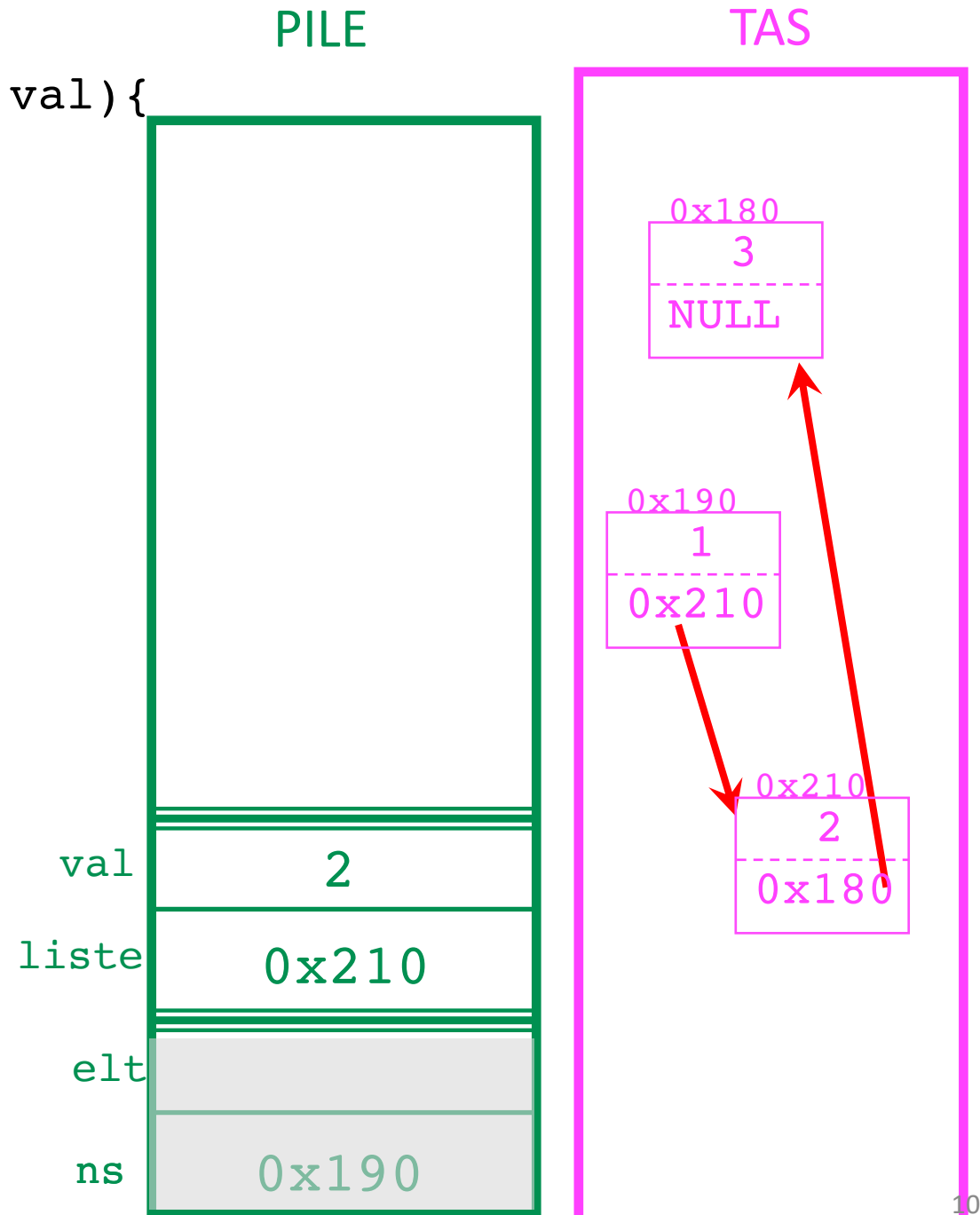
```
int main(){  
    int_list ns, elt;  
    ns=cons(1,cons(2,cons(3,NULL)));  
    elt= rechercher(ns,2);  
    if(elt!=NULL)  
        printf("2 est présent\n »);  
    return 0;  
}
```



Rechercher un élément dans une liste

```
int_list rechercher(int_list liste, int val){
    while(liste !=NULL){
        if(liste->car==val)
            return liste;
        else
            liste=liste->cdr;
    }
    return NULL,
}
```

```
int main(){
    int_list ns, elt;
    ns=cons(1,cons(2,cons(3,NULL)));
    elt= rechercher(ns,2);
    if(elt!=NULL)
        printf("2 est présent\n »);
    return 0;
}
```

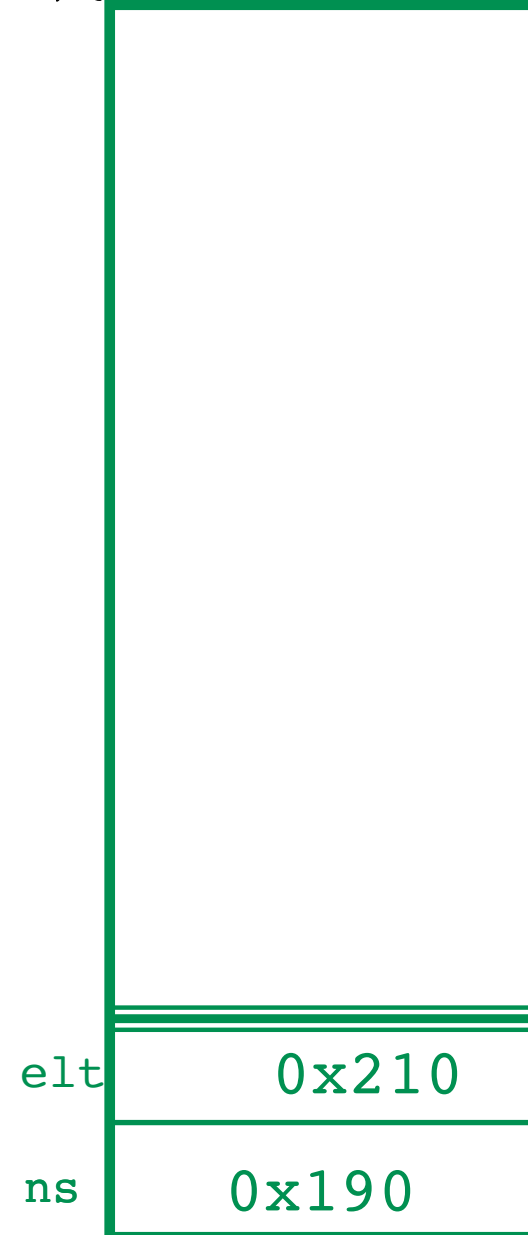


Rechercher un élément dans une liste

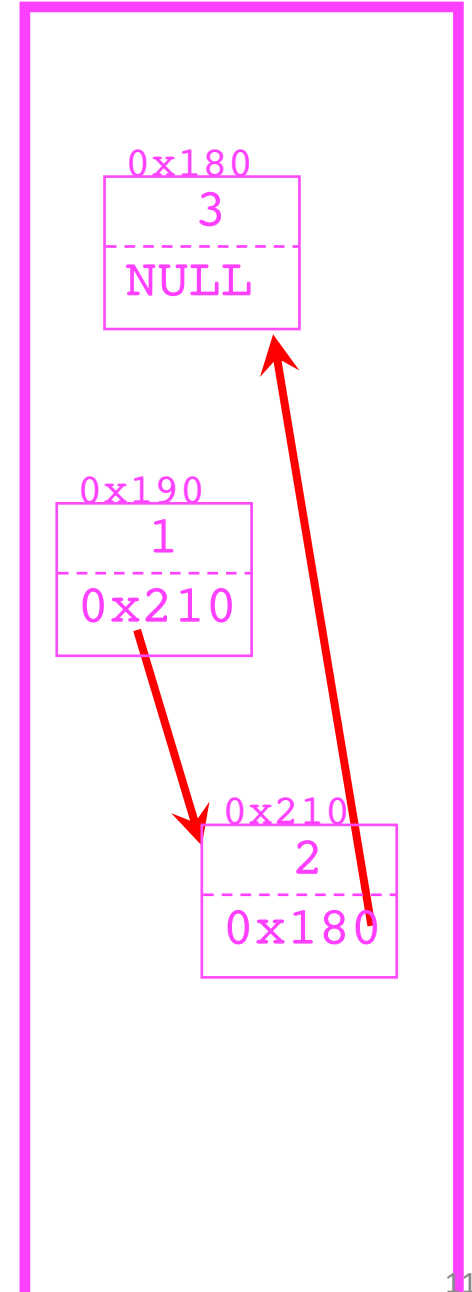
```
int_list rechercher(int_list liste, int val){
    while(liste !=NULL){
        if(liste->car==val)
            return liste;
        else
            liste=liste->cdr;
    }
    return NULL,
}
```

```
int main(){
    int_list ns, elt;
    ns=cons(1,cons(2,cons(3,NULL)));
    elt= rechercher(ns,2);
    if(elt!=NULL)
        printf("2 est présent\n »);
    return 0;
}
```

PILE



TAS



Rechercher un élément dans une liste

```
int_list rechercher(int_list liste, int val){
    while(liste !=NULL){
        if(liste->car==val)
            return liste;
        else
            liste=liste->cdr;
    }
    return NULL,
}
```

Jeu de test:

```
int main(){
    int_list ns, elt;
    ns=cons(1,cons(2,cons(3,NULL)));
    assert(rechercher(ns, 1)!=NULL);
    assert( rechercher(ns,2 )!=NULL);
    assert( rechercher(ns,3 )!=NULL);
    assert( rechercher(ns,4 )==NULL);
    return 0;
}
```

Cela marche t-il si :

- l'élément à trouver est en 1^{er} ?
- l'élément à trouver est en dernier ?
- l'élément à trouver est au milieu ?
- l'élément à trouver est absent ?