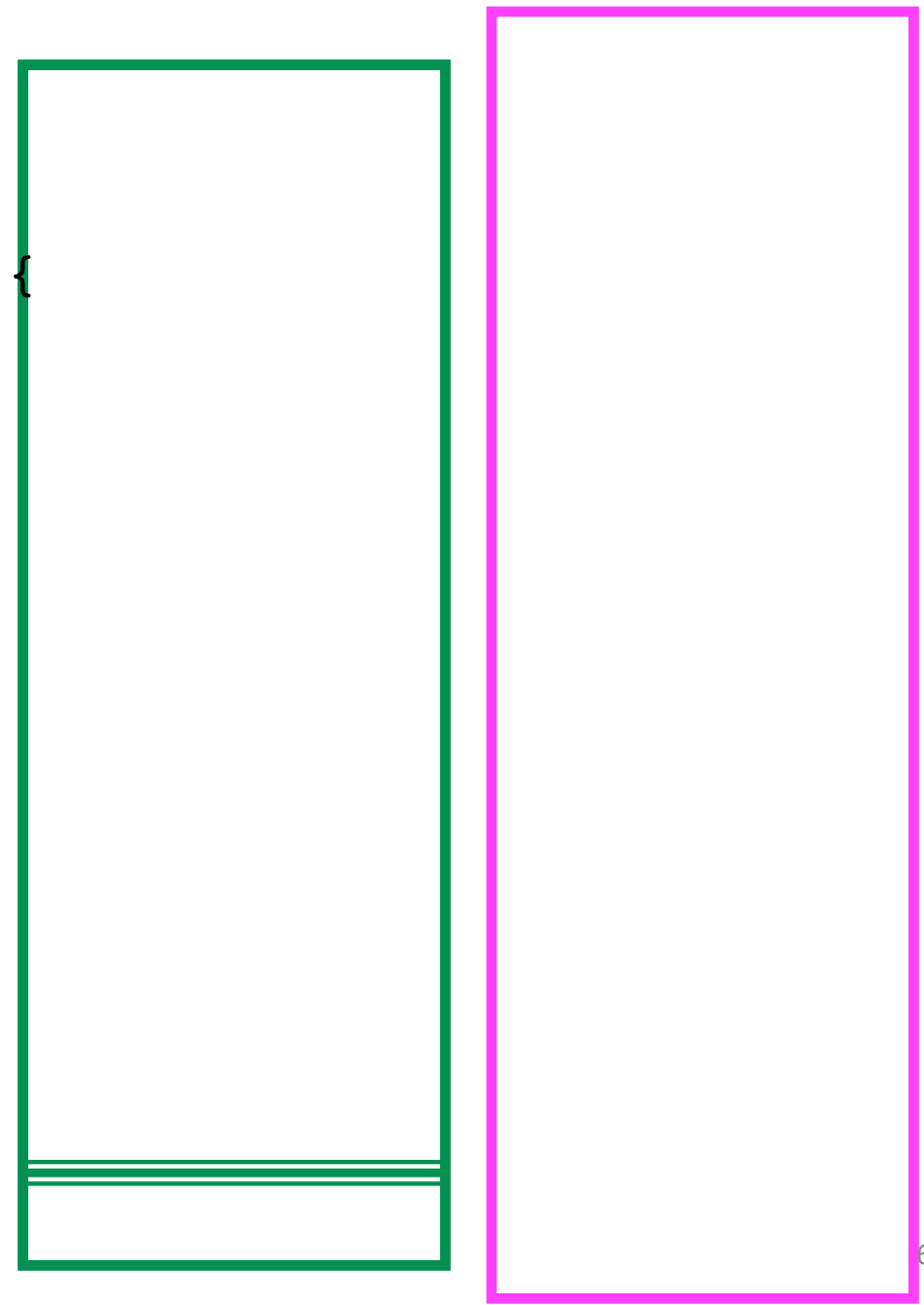


```
typedef struct _cellule_t cellule_t;
struct _cellule_t{
    int donnee;
    cellule_t *suivant;
};
cellule_t *cons(int val, cellule_t *pListe){
    cellule_t *el;

    el = malloc(sizeof(cellule_t));
    if (el == NULL) return NULL;
    el->donnee=val;
    el->suivant = pListe;
    return el;
}
int main(){
    → cellule_t *ns;
    ns=cons(3,ns);
    ns=cons(2,ns);
    ns=cons(1,ns);
    return 0;
}
```

PILE

TAS



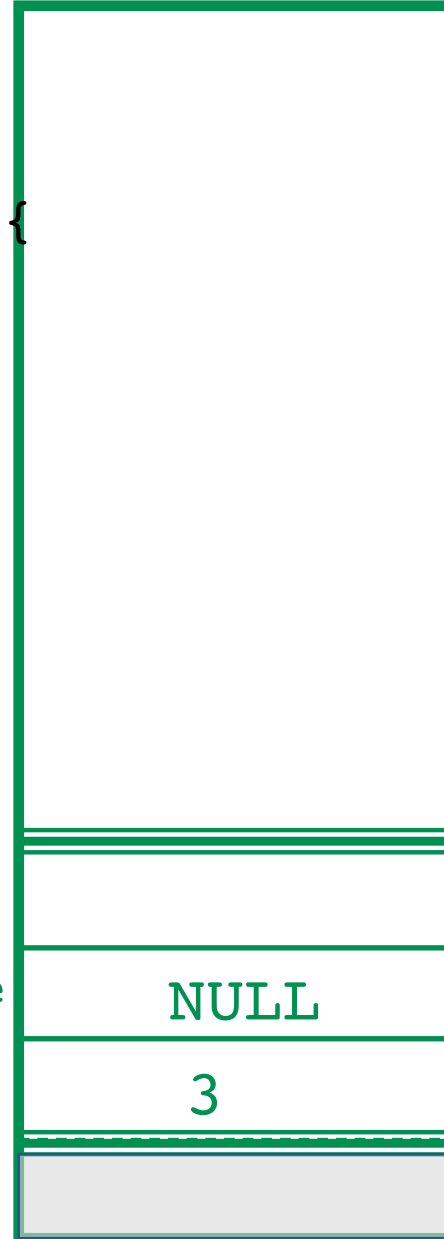
```
typedef struct _cellule_t cellule_t;
struct _cellule_t{
    int donnee;
    cellule_t *suivant;
};
cellule_t *cons(int val, cellule_t *pListe){
    cellule_t *el;

    el = malloc(sizeof(cellule_t));
    if (el == NULL) return NULL;
    el->donnee=val;
    el->suivant = pListe;
    return el;
}
int main(){
    cellule_t *ns;
    ns=cons(3,ns);
    ns=cons(2,ns);
    ns=cons(1,ns);
    return 0;
}
```

PILE

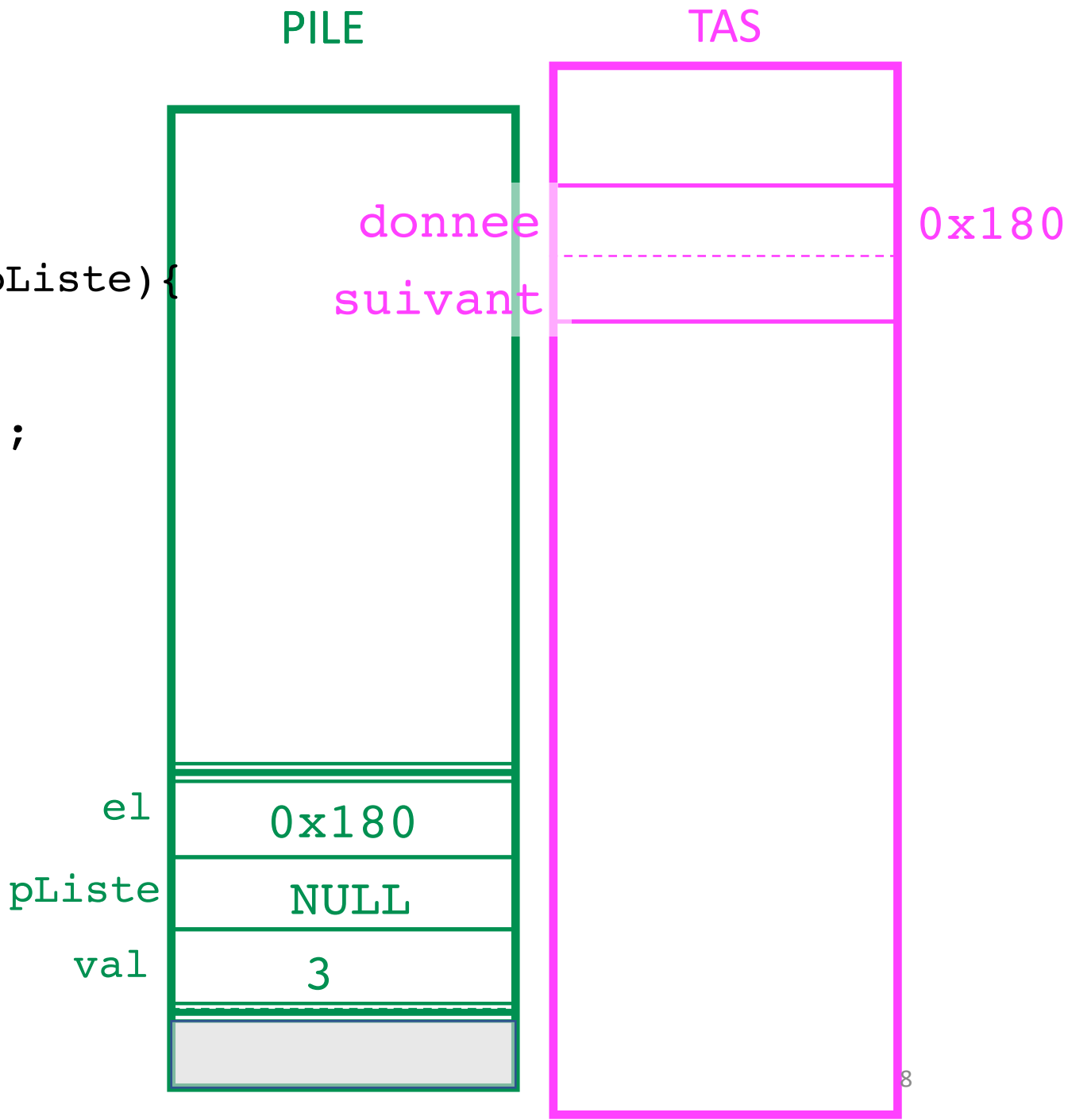
TAS

el  
pListe  
val



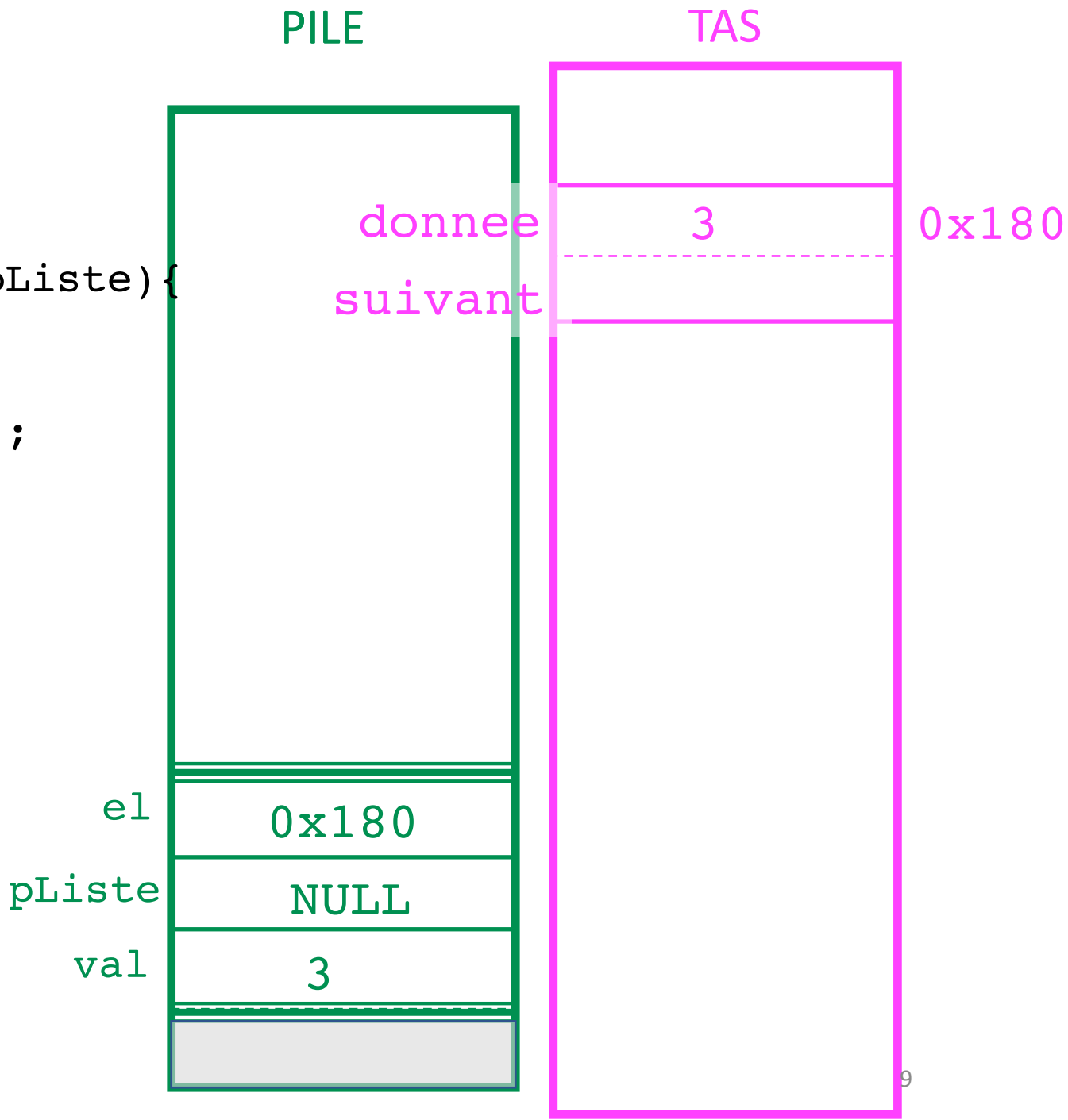
```
typedef struct _cellule_t cellule_t;
struct _cellule_t{
    int donnee;
    cellule_t *suivant;
};
cellule_t *cons(int val, cellule_t *pListe){
    cellule_t *el;

    el = malloc(sizeof(cellule_t));
    if (el == NULL) return NULL;
    el->donnee=val;
    el->suivant = pListe;
    return el;
}
int main(){
    cellule_t *ns;
    ns=cons(3,ns);
    ns=cons(2,ns);
    ns=cons(1,ns);
    return 0;
}
```



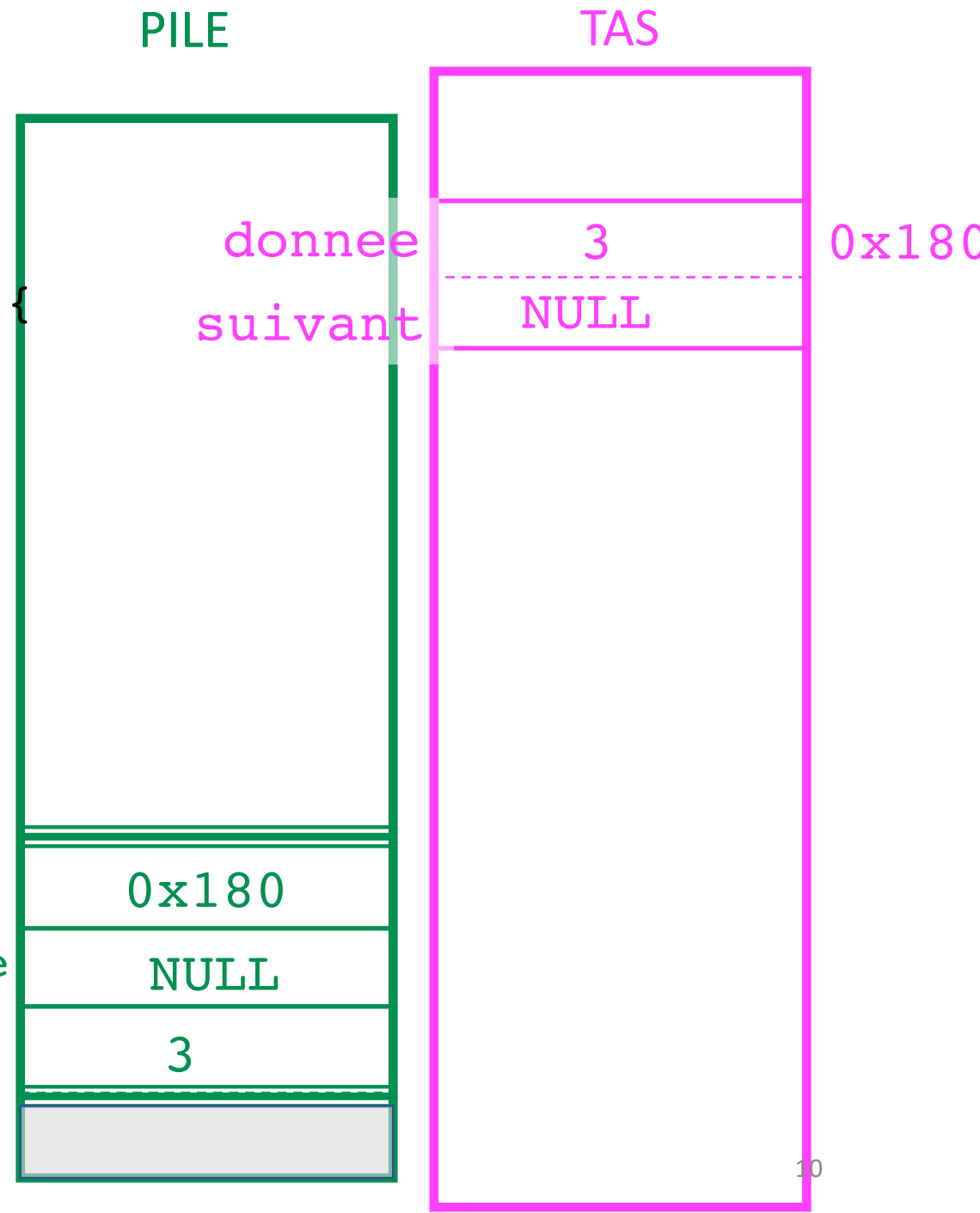
```
typedef struct _cellule_t cellule_t;
struct _cellule_t{
    int donnee;
    cellule_t *suivant;
};
cellule_t *cons(int val, cellule_t *pListe){
    cellule_t *el;

    el = malloc(sizeof(cellule_t));
    if (el == NULL) return NULL;
    el->donnee=val;
    el->suivant = pListe;
    return el;
}
int main(){
    cellule_t *ns;
    ns=cons(3,ns);
    ns=cons(2,ns);
    ns=cons(1,ns);
    return 0;
}
```



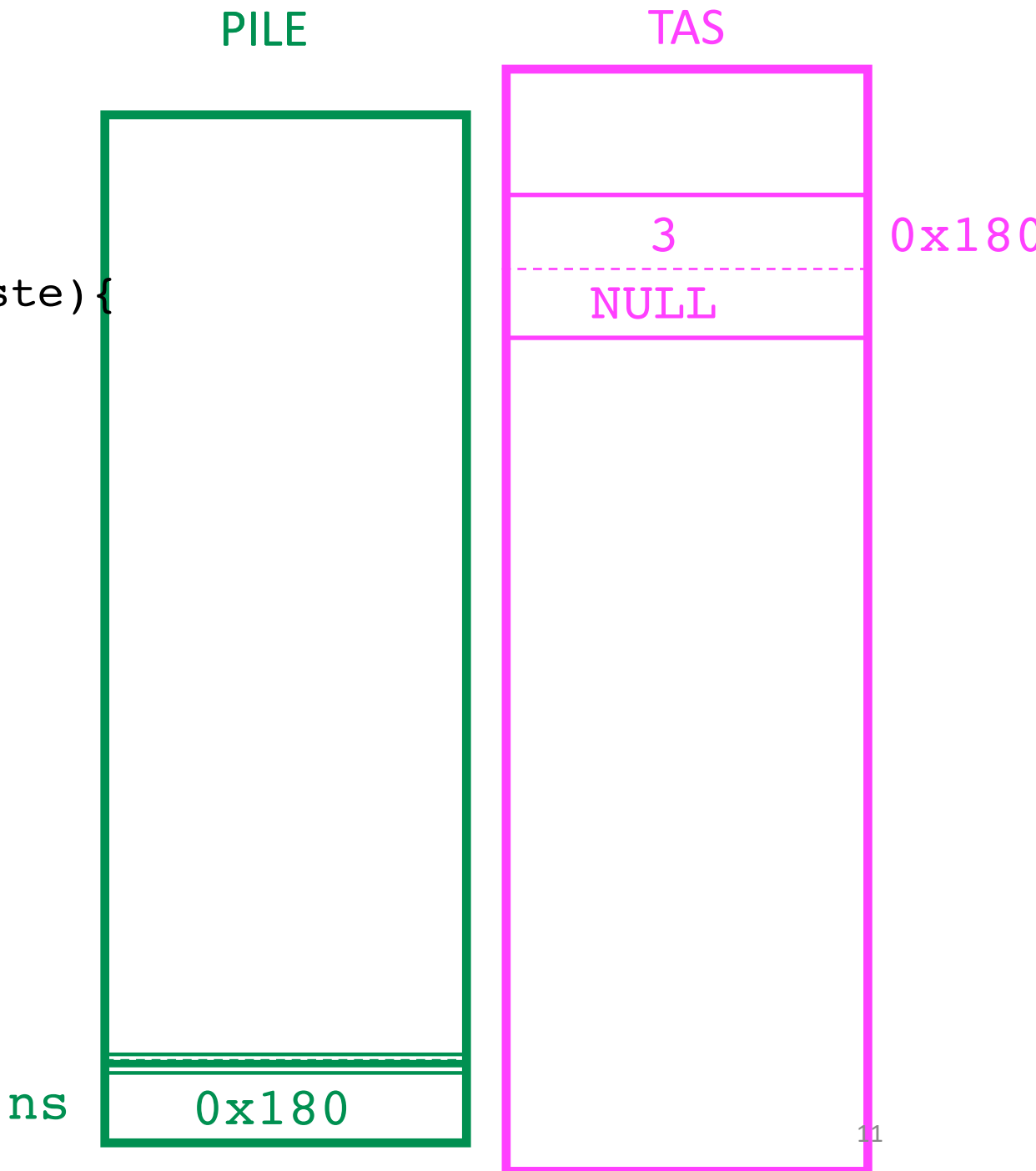
```
typedef struct _cellule_t cellule_t;
struct _cellule_t{
    int donnee;
    cellule_t *suivant;
};
cellule_t *cons(int val, cellule_t *pListe){
    cellule_t *el;

    el = malloc(sizeof(cellule_t));
    if (el == NULL) return NULL;
    el->donnee=val;
    el->suivant = pListe;
    return el;
}
int main(){
    cellule_t *ns;
    ns=cons(3,ns);
    ns=cons(2,ns);
    ns=cons(1,ns);
    return 0;
}
```



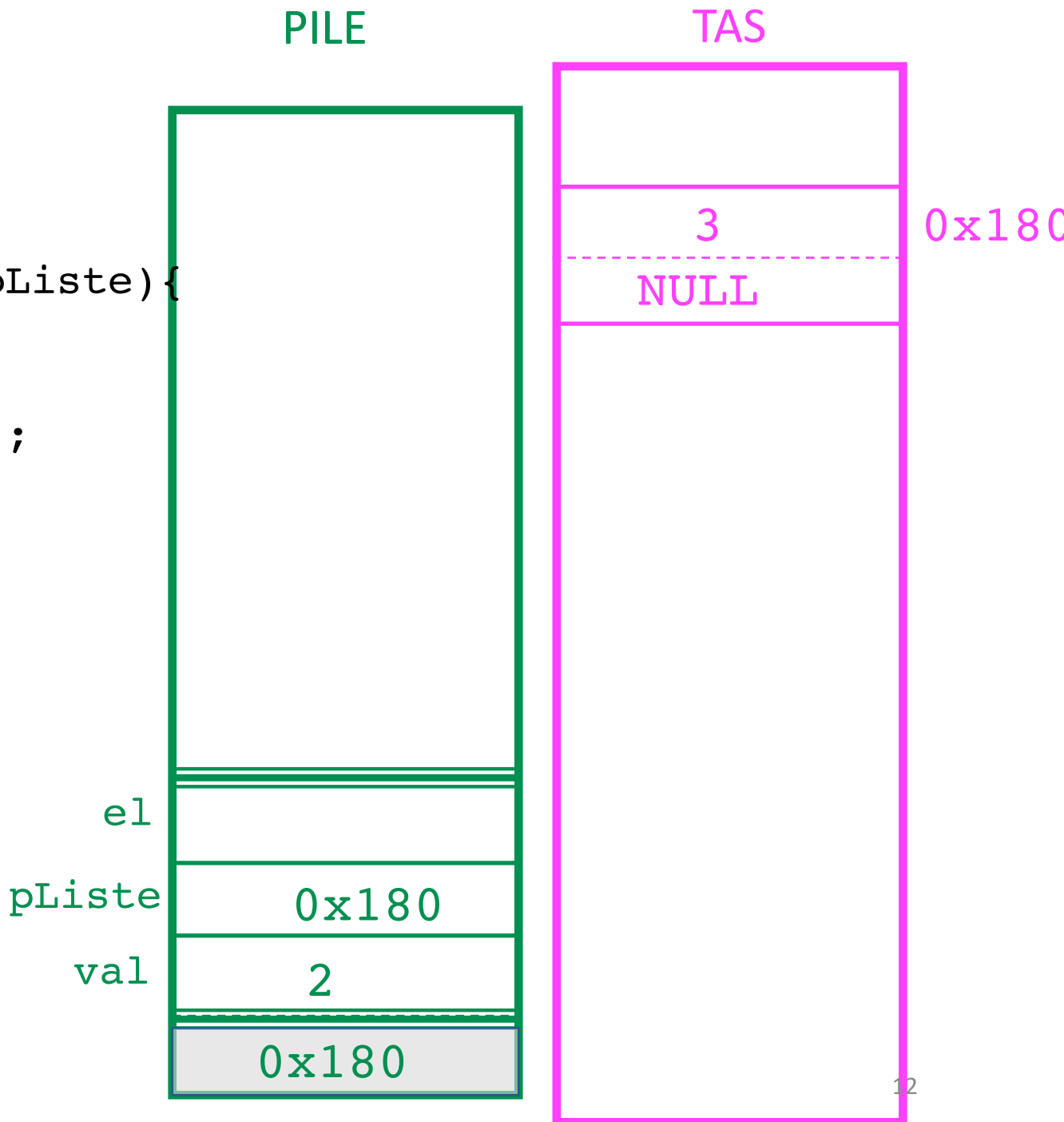
```
typedef struct _cellule_t cellule_t;
struct _cellule_t{
    int donnee;
    cellule_t *suivant;
};
cellule_t *cons(int val, cellule_t *pListe){
    cellule_t *el;

    el = malloc(sizeof(cellule_t));
    if (el == NULL) return NULL;
    el->donnee=val;
    el->suivant = pListe;
    return el;
}
int main(){
    cellule_t *ns;
    ns=cons(3,ns);
    ns=cons(2,ns);
    ns=cons(1,ns);
    return 0;
}
```



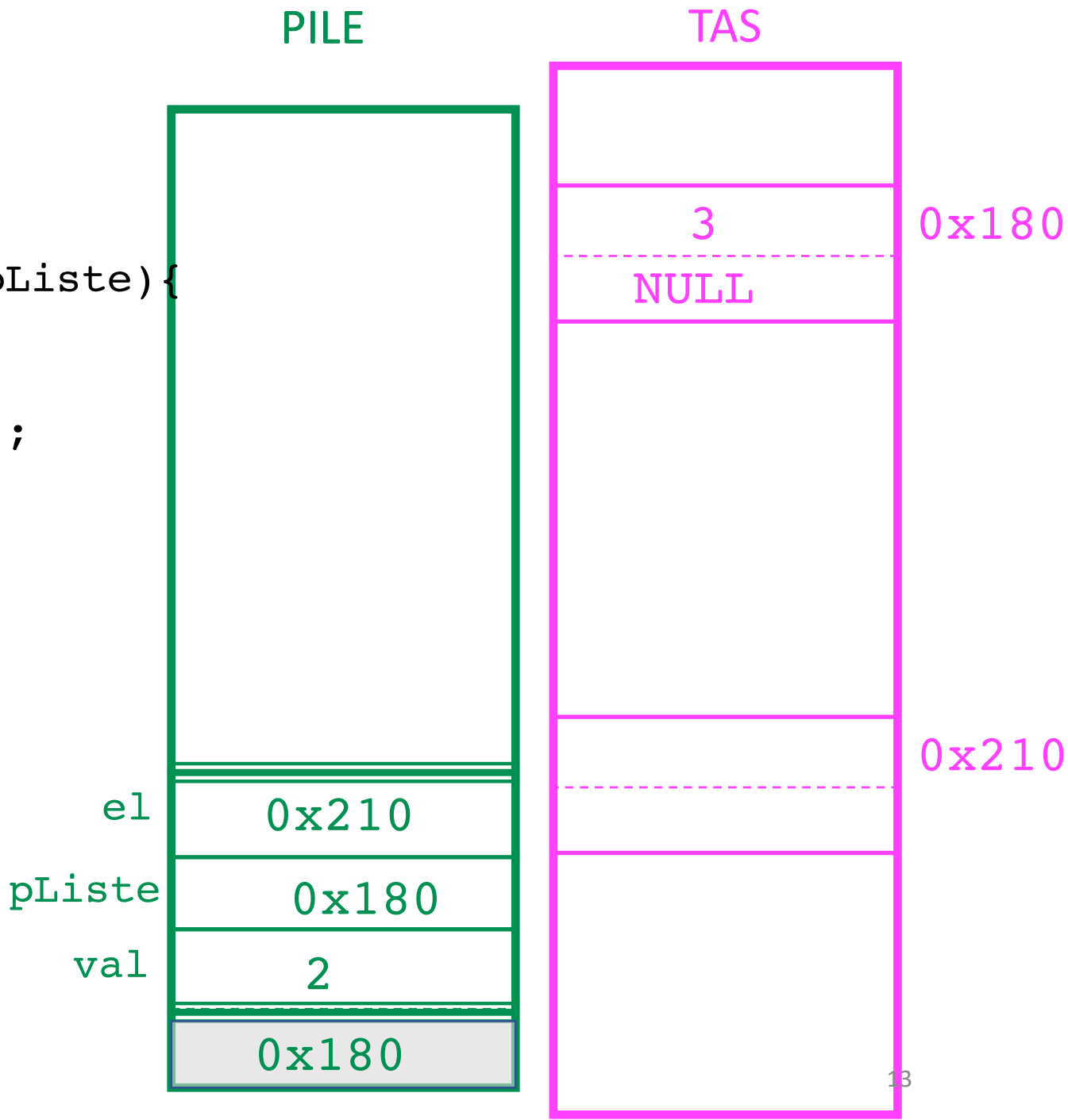
```
typedef struct _cellule_t cellule_t;
struct _cellule_t{
    int donnee;
    cellule_t *suivant;
};
cellule_t *cons(int val, cellule_t *pListe){
    cellule_t *el;

    el = malloc(sizeof(cellule_t));
    if (el == NULL) return NULL;
    el->donnee=val;
    el->suivant = pListe;
    return el;
}
int main(){
    cellule_t *ns;
    ns=cons(3,ns);
    ns=cons(2,ns);
    ns=cons(1,ns);
    return 0;
}
```



```
typedef struct _cellule_t cellule_t;
struct _cellule_t{
    int donnee;
    cellule_t *suivant;
};
cellule_t *cons(int val, cellule_t *pListe){
    cellule_t *el;

    el = malloc(sizeof(cellule_t));
    if (el == NULL) return NULL;
    el->donnee=val;
    el->suivant = pListe;
    return el;
}
int main(){
    cellule_t *ns;
    ns=cons(3,ns);
    ns=cons(2,ns);
    ns=cons(1,ns);
    return 0;
}
```





```

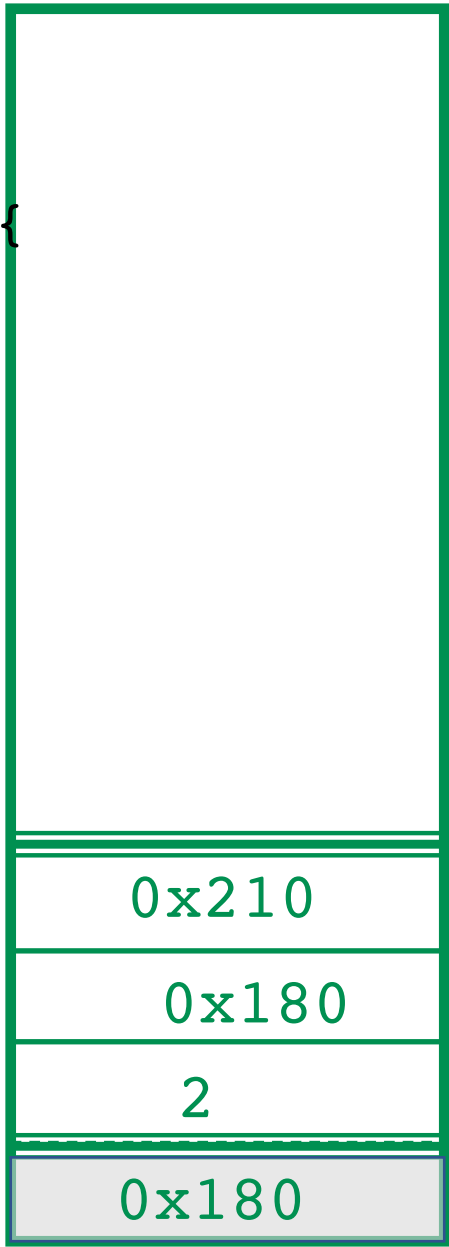
typedef struct _cellule_t cellule_t;
struct _cellule_t{
    int donnee;
    cellule_t *suivant;
};
cellule_t *cons(int val, cellule_t *pListe){
    cellule_t *el;

    el = malloc(sizeof(cellule_t));
    if (el == NULL) return NULL;
    el->donnee=val;
    el->suivant = pListe;
    return el;
}
int main(){
    cellule_t *ns;
    ns=cons(3,ns);
    ns=cons(2,ns);
    ns=cons(1,ns);
    return 0;
}

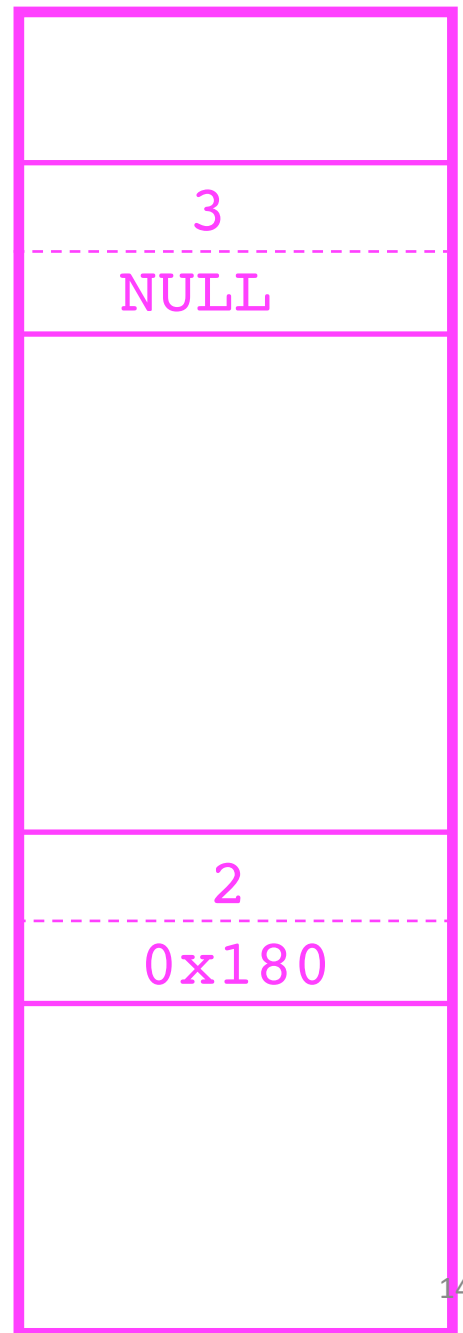
```



PILE



TAS

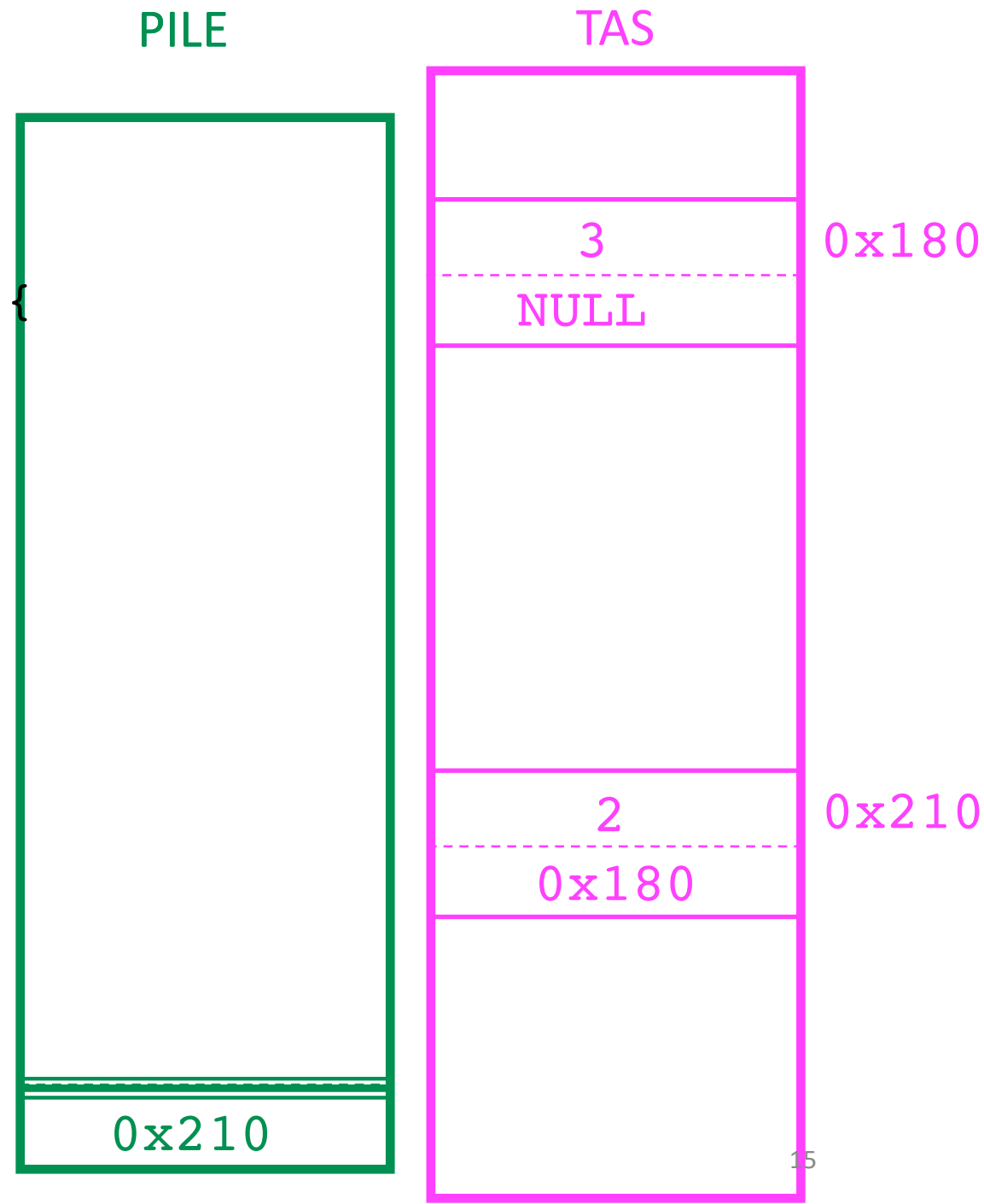


0x180

0x210

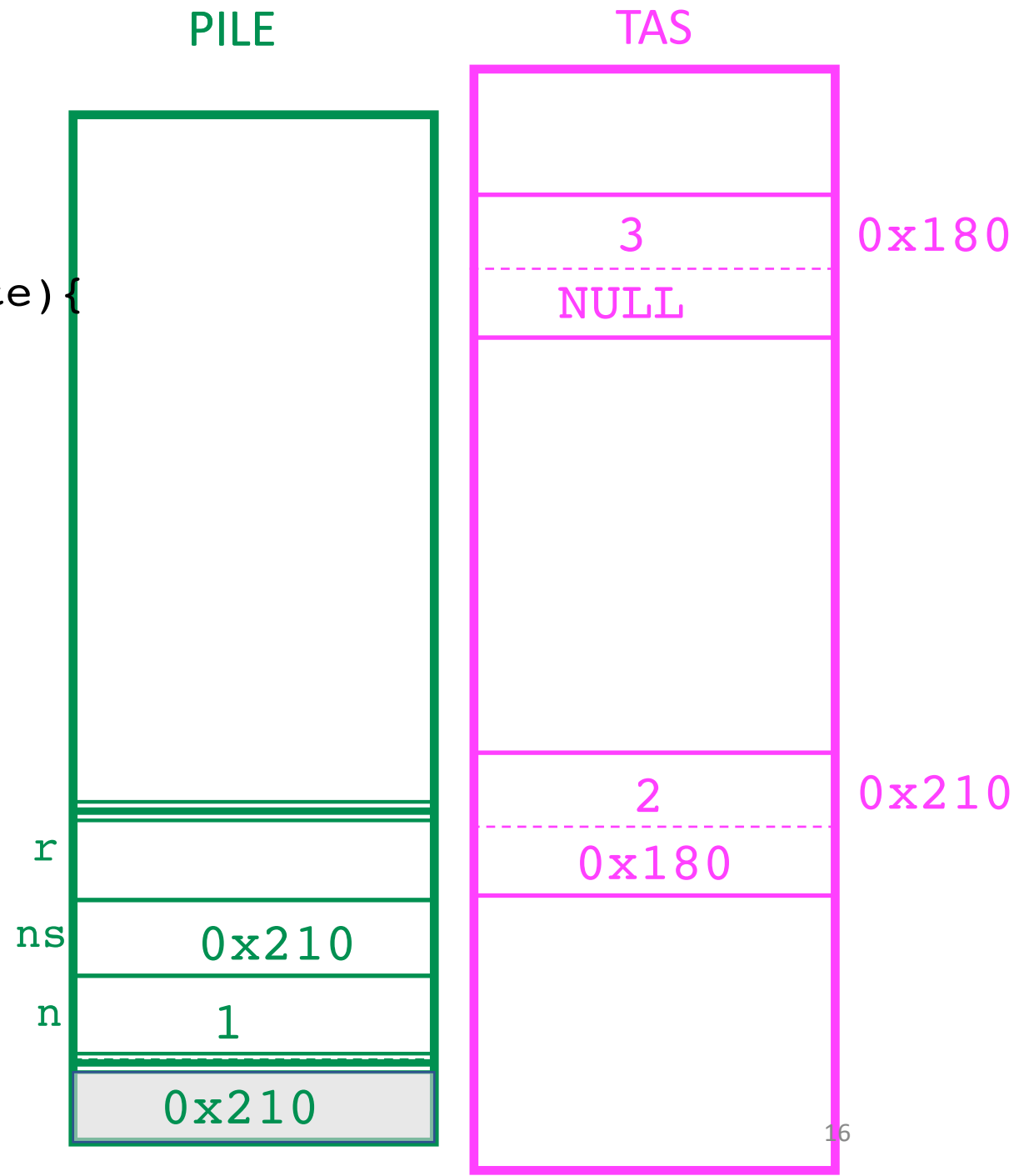
```
typedef struct _cellule_t cellule_t;
struct _cellule_t{
    int donnee;
    cellule_t *suivant;
};
cellule_t *cons(int val, cellule_t *pListe){
    cellule_t *el;

    el = malloc(sizeof(cellule_t));
    if (el == NULL) return NULL;
    el->donnee=val;
    el->suivant = pListe;
    return el;
}
int main(){
    cellule_t *ns;
    ns=cons(3,ns);
    ns=cons(2,ns);
    ns=cons(1,ns);
    return 0;
}
```



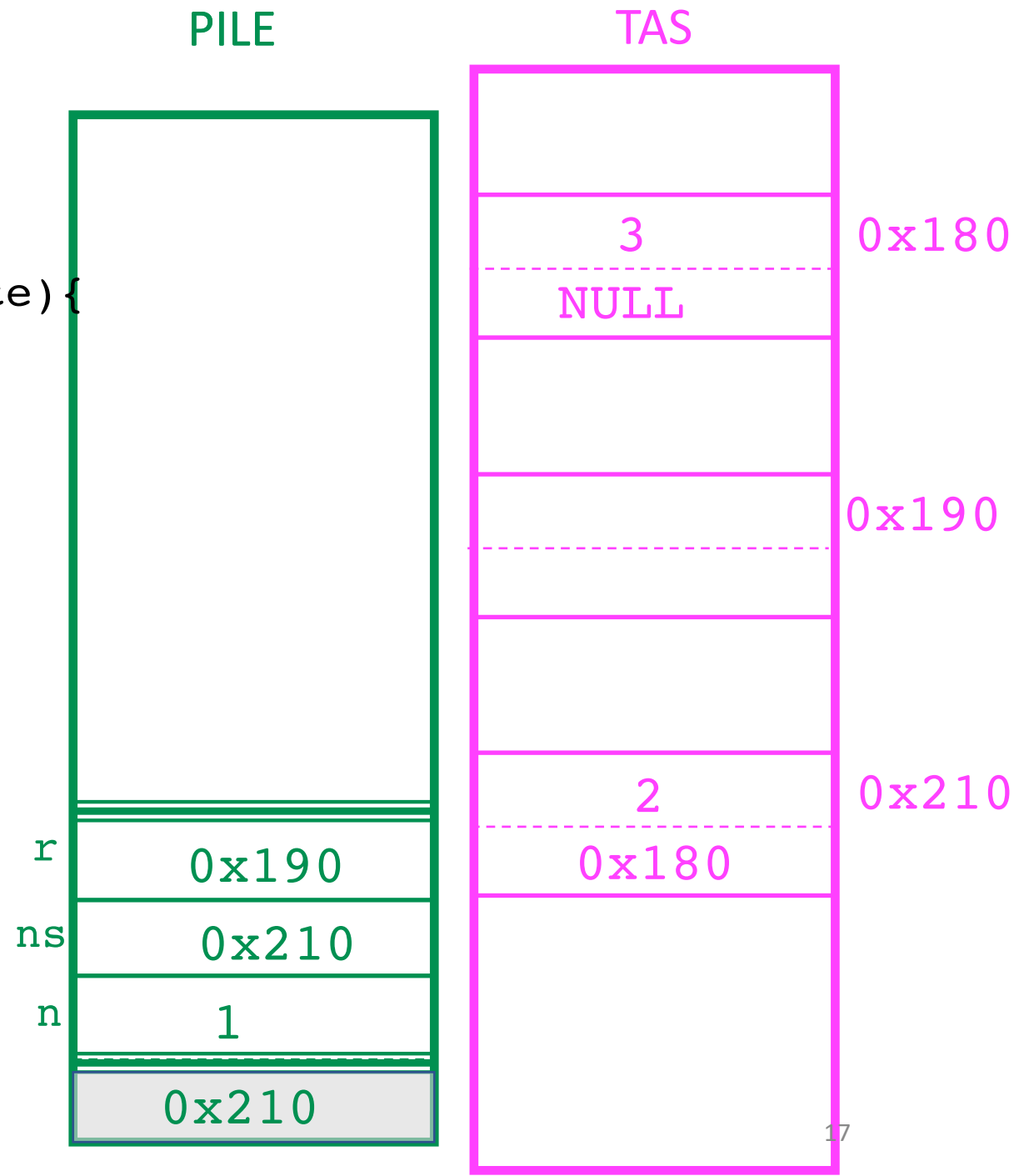
```
typedef struct _cellule_t cellule_t;
struct _cellule_t{
    int donnee;
    cellule_t *suivant;
};
cellule_t *cons(int val, cellule_t *pListe){
    cellule_t *el;

    el = malloc(sizeof(cellule_t));
    if (el == NULL) return NULL;
    el->donnee=val;
    el->suivant = pListe;
    return el;
}
int main(){
    cellule_t *ns;
    ns=cons(3,ns);
    ns=cons(2,ns);
    ns=cons(1,ns);
    return 0;
}
```



```
typedef struct _cellule_t cellule_t;
struct _cellule_t{
    int donnee;
    cellule_t *suivant;
};
cellule_t *cons(int val, cellule_t *pListe){
    cellule_t *el;

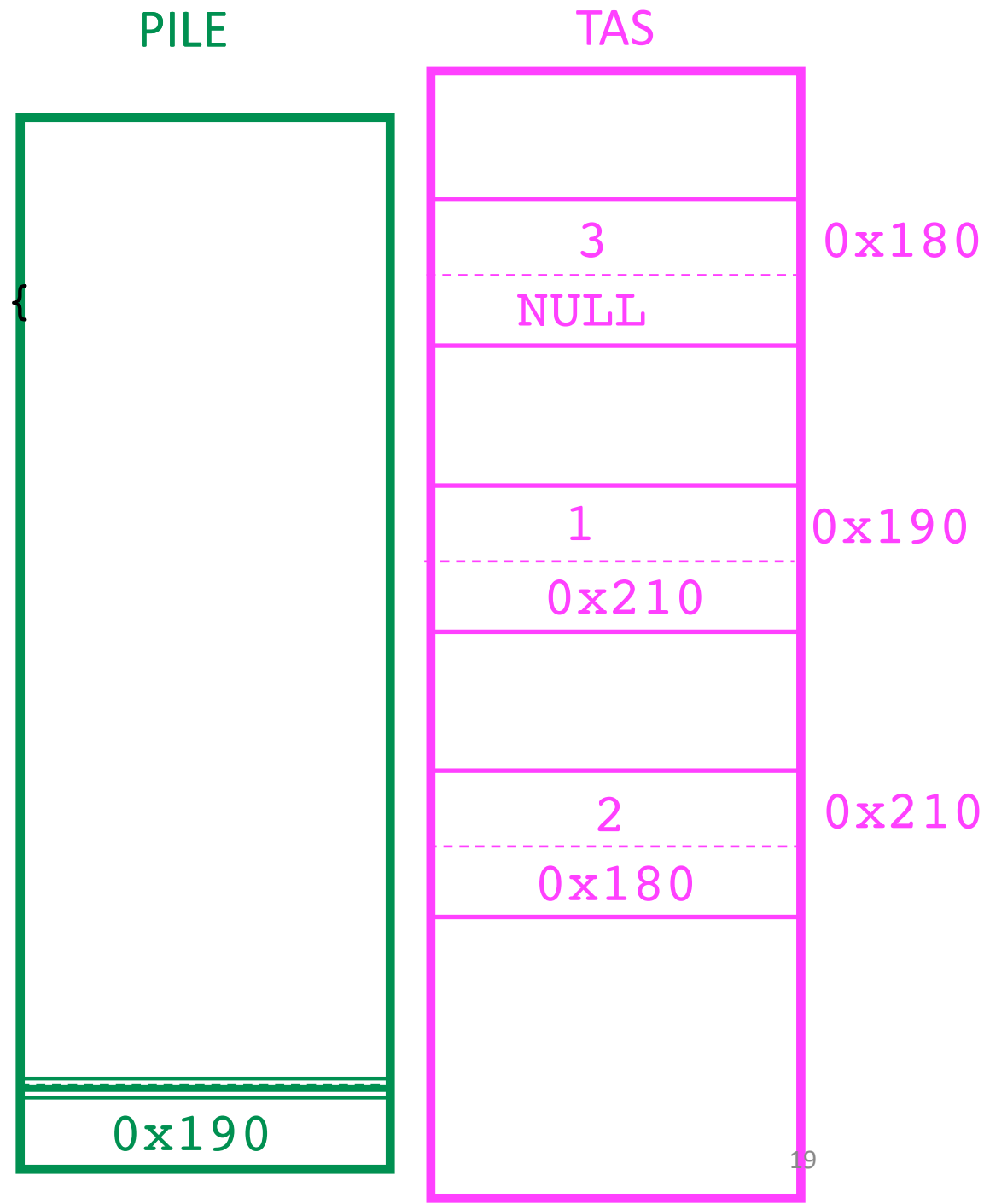
    el = malloc(sizeof(cellule_t));
    if (el == NULL) return NULL;
    el->donnee=val;
    el->suivant = pListe;
    return el;
}
int main(){
    cellule_t *ns;
    ns=cons(3,ns);
    ns=cons(2,ns);
    ns=cons(1,ns);
    return 0;
}
```





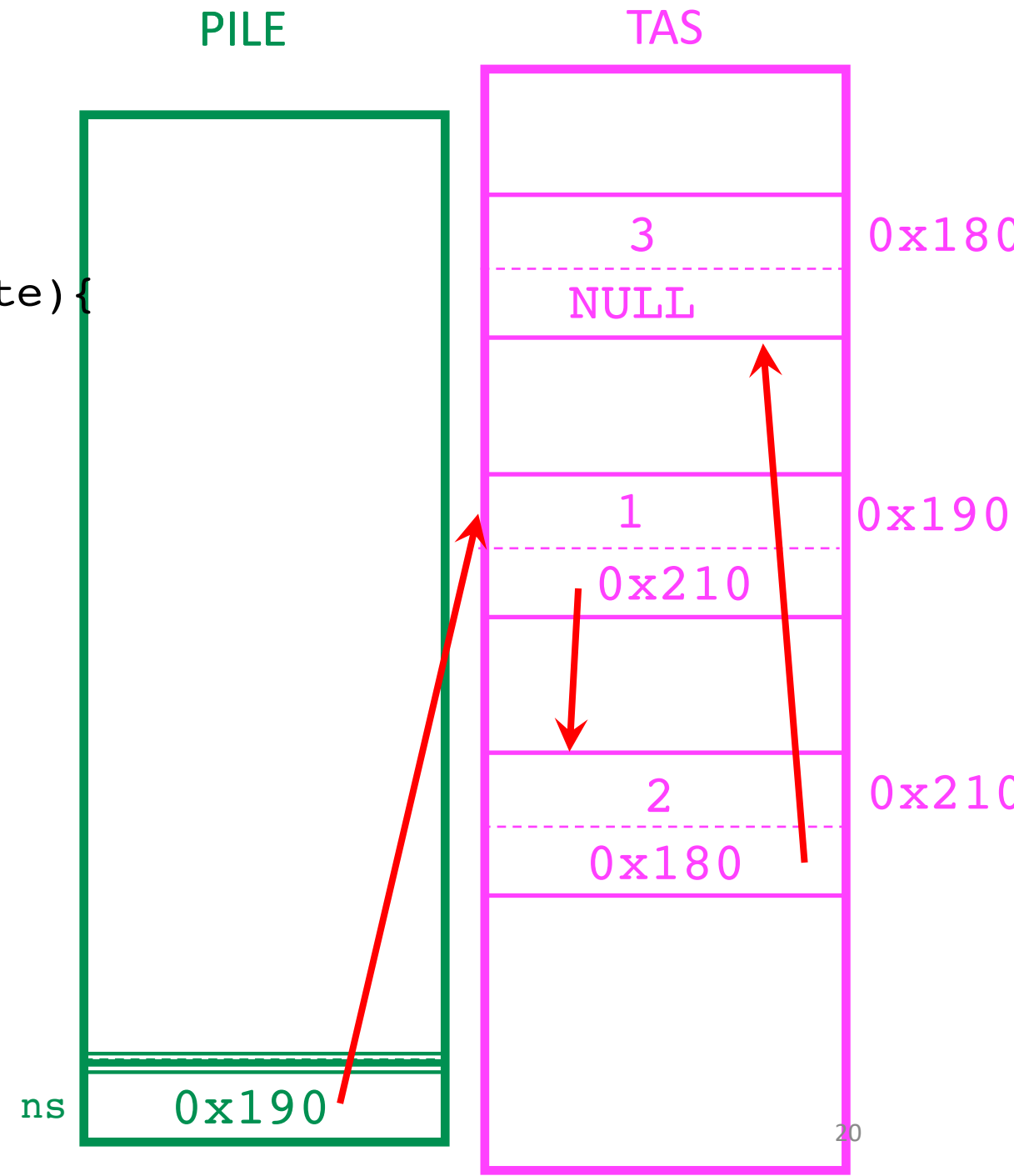
```
typedef struct _cellule_t cellule_t;
struct _cellule_t{
    int donnee;
    cellule_t *suivant;
};
cellule_t *cons(int val, cellule_t *pListe){
    cellule_t *el;

    el = malloc(sizeof(cellule_t));
    if (el == NULL) return NULL;
    el->donnee=val;
    el->suivant = pListe;
    return el;
}
int main(){
    cellule_t *ns;
    ns=cons(3,ns);
    ns=cons(2,ns);
    ns=cons(1,ns);
    return 0;
}
```



```
typedef struct _cellule_t cellule_t;
struct _cellule_t{
    int donnee;
    cellule_t *suivant;
};
cellule_t *cons(int val, cellule_t *pListe){
    cellule_t *el;

    el = malloc(sizeof(cellule_t));
    if (el == NULL) return NULL;
    el->donnee=val;
    el->suivant = pListe;
    return el;
}
int main(){
    cellule_t *ns;
    ns=cons(3,ns);
    ns=cons(2,ns);
    ns=cons(1,ns);
    return 0;
}
```



```

typedef struct _cellule_t cellule_t;
struct _cellule_t{
    int donnee;
    cellule_t *suivant;
};
cellule_t *cons(int val, cellule_t *pListe){
    cellule_t *el;

    el = malloc(sizeof(cellule_t));
    if (el == NULL) return NULL;
    el->donnee=val;
    el->suivant = pListe;
    return el;
}
int main(){
    cellule_t *ns;
    ns=cons(3,ns);
    ns=cons(2,ns);
    ns=cons(1,ns);
    return 0;
}

```



ns

PILE

TAS

