

# Matrices Dinámicas

Rafael Palacios 2005

# Matrices dinámicas

```
/* Programa asignar una matriz dinámica */
#include <stdio.h>

int main(void)
{
    int f; /* número de filas */
    int c; /* número de columnas */
    double **mat; /* puntero de puntero */
    int i; /* contador */

    /* obtengo f y c */
    f=3;
    c=7;

    /* Asigno memoria (falta el control de errores)*/
    mat=(double **)calloc(f, sizeof(double *));
    for(i=0; i<f; i++) {
        mat[i]=calloc(c, sizeof(double));
    }

    <... sigue...>
```

# Matrices dinámicas

```
/* Programa asignar una matriz dinámica */
```

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    int f; /* número de filas */
```

```
    int c; /* número de columnas */
```

```
    double **mat; /* puntero de puntero */
```

```
    int i; /* contador */
```

```
    /* obtengo f y c */
```

```
    f=3;
```

```
    c=7;
```

```
    /* Asigno memoria (falta el control de errores)*/
```

```
    mat=(double **)calloc(f, sizeof(double *));
```

```
    for(i=0; i<f; i++) {
```

```
        mat[i]=calloc(c, sizeof(double));
```

```
    }
```

```
    <... sigue... >
```

f	???
c	???
mat	???
i	???

# Matrices dinámicas

```
/* Programa asignar una matriz dinámica */
```

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    int f; /* número de filas */
```

```
    int c; /* número de columnas */
```

```
    double **mat; /* puntero de puntero */
```

```
    int i; /* contador */
```

```
    /* obtengo f y c */
```

```
    f=3;
```

```
    c=7;
```

```
    /* Asigno memoria (falta el control de errores)*/
```

```
    mat=(double **)calloc(f,sizeof(double *));
```

```
    for(i=0; i<f; i++) {
```

```
        mat[i]=calloc(c,sizeof(double));
```

```
    }
```

```
    <... sigue... >
```

f	3
c	???
mat	???
i	???

# Matrices dinámicas

```
/* Programa asignar una matriz dinámica */
```

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    int f; /* número de filas */
```

```
    int c; /* número de columnas */
```

```
    double **mat; /* puntero de puntero */
```

```
    int i; /* contador */
```

```
    /* obtengo f y c */
```

```
    f=3;
```

```
    c=7;
```

```
    /* Asigno memoria (falta el control de errores)*/
```

```
    mat=(double **)calloc(f, sizeof(double *));
```

```
    for(i=0; i<f; i++) {
```

```
        mat[i]=calloc(c, sizeof(double));
```

```
    }
```

```
    <... sigue... >
```

f	3
c	7
mat	???
i	???



# Matrices dinámicas

```
/* Programa asignar una matriz dinámica */
```

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    int f; /* número de filas */
```

```
    int c; /* número de columnas */
```

```
    double **mat; /* puntero de puntero */
```

```
    int i; /* contador */
```

```
    /* obtengo f y c */
```

```
    f=3;
```

```
    c=7;
```

```
    /* Asigno memoria (falta el control de errores)*/
```

```
    mat=(double **)calloc(f,sizeof(double *));
```

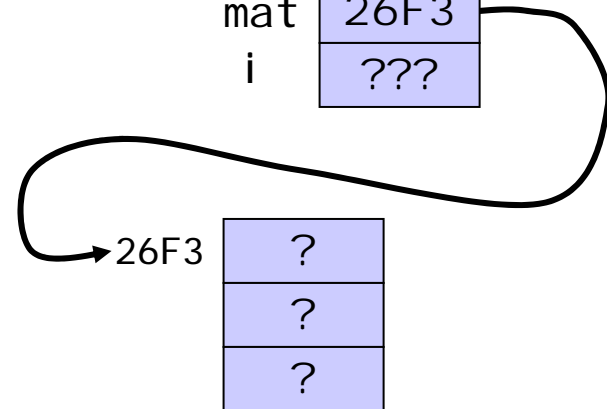
```
    for(i=0; i<f; i++) {
```

```
        mat[i]=calloc(c,sizeof(double));
```

```
    }
```

```
    <... si gue... >
```

f	3
c	7
mat	26F3
i	???



# Matrices dinámicas

```
/* Programa asignar una matriz dinámica */
```

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    int f; /* número de filas */
```

```
    int c; /* número de columnas */
```

```
    double **mat; /* puntero de puntero */
```

```
    int i; /* contador */
```

```
    /* obtengo f y c */
```

```
    f=3;
```

```
    c=7;
```

```
    /* Asigno memoria (falta el control de errores)*/
```

```
    mat=(double **)calloc(f,sizeof(double *));
```

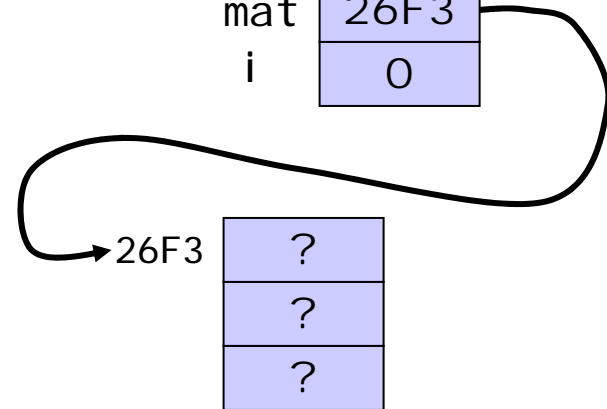
```
    for(i=0; i<f; i++) {
```

```
        mat[i]=(double *)calloc(c,sizeof(double));
```

```
    }
```

```
    <... si gue... >
```

f	3
c	7
mat	26F3
i	0



# Matrices dinámicas

```
/* Programa asignar una matriz dinámica */
```

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    int f; /* número de filas */
```

```
    int c; /* número de columnas */
```

```
    double **mat; /* puntero de puntero */
```

```
    int i; /* contador */
```

```
    /* obtengo f y c */
```

```
    f=3;
```

```
    c=7;
```

```
    /* Asigno memoria (falta el control de errores)*/
```

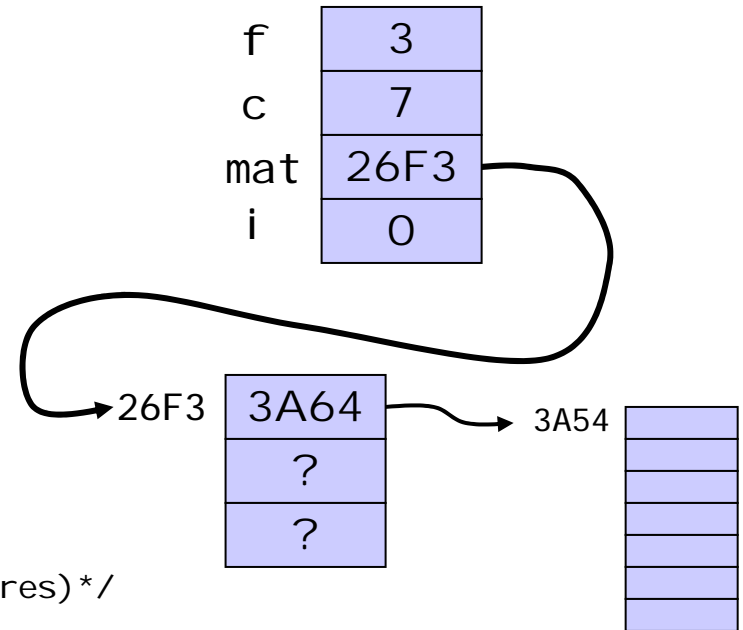
```
    mat=(double **)calloc(f,sizeof(double *));
```

```
    for(i=0; i<f; i++) {
```

```
        mat[i]=calloc(c,sizeof(double));
```

```
    }
```

```
<... si gue... >
```





# Matrices dinámicas

```
/* Programa asignar una matriz dinámica */
```

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    int f; /* número de filas */
```

```
    int c; /* número de columnas */
```

```
    double **mat; /* puntero de puntero */
```

```
    int i; /* contador */
```

```
    /* obtengo f y c */
```

```
    f=3;
```

```
    c=7;
```

```
    /* Asigno memoria (falta el control de errores)*/
```

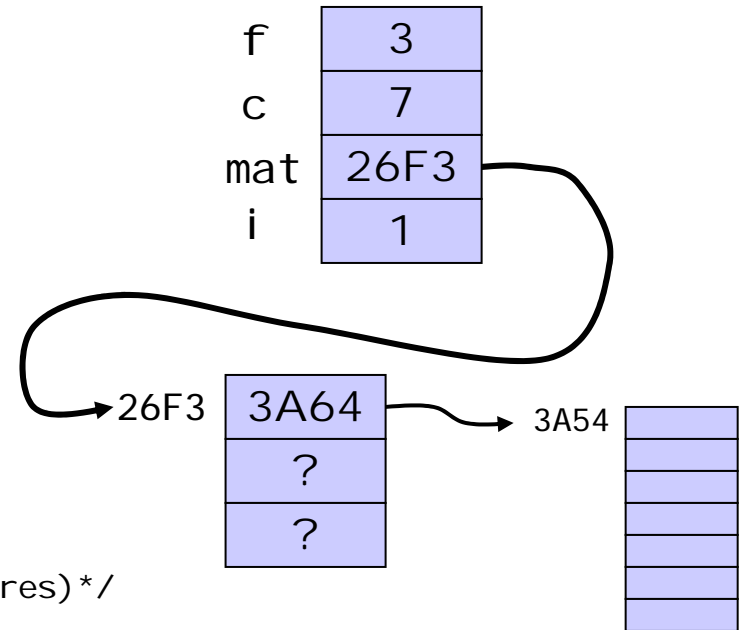
```
    mat=(double **)calloc(f,sizeof(double *));
```

```
    for(i=0; i<f; i++) {
```

```
        mat[i]=(double *)calloc(c,sizeof(double));
```

```
    }
```

```
    <... si gue... >
```



# Matrices dinámicas

```
/* Programa asignar una matriz dinámica */
```

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    int f; /* número de filas */
```

```
    int c; /* número de columnas */
```

```
    double **mat; /* puntero de puntero */
```

```
    int i; /* contador */
```

```
    /* obtengo f y c */
```

```
    f=3;
```

```
    c=7;
```

```
    /* Asigno memoria (falta el control de errores)*/
```

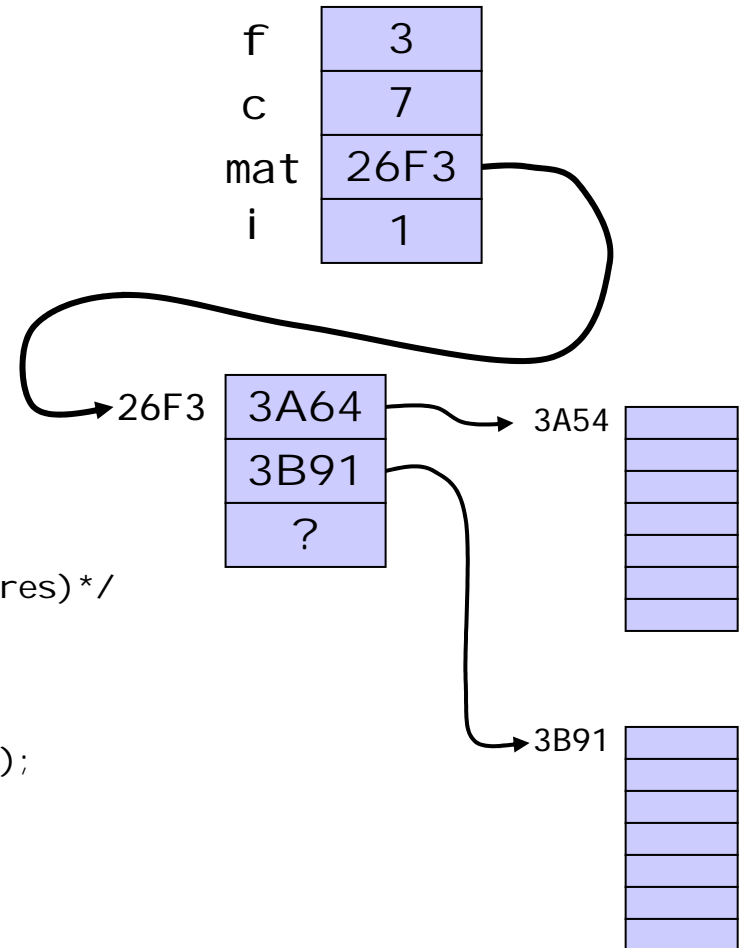
```
    mat=(double **)calloc(f,sizeof(double *));
```

```
    for(i=0; i<f; i++) {
```

```
        mat[i]=(double *)calloc(c,sizeof(double));
```

```
    }
```

```
    <... si gue... >
```



# Matrices dinámicas

```
/* Programa asignar una matriz dinámica */
```

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    int f; /* número de filas */
```

```
    int c; /* número de columnas */
```

```
    double **mat; /* puntero de puntero */
```

```
    int i; /* contador */
```

```
    /* obtengo f y c */
```

```
    f=3;
```

```
    c=7;
```

```
    /* Asigno memoria (falta el control de errores)*/
```

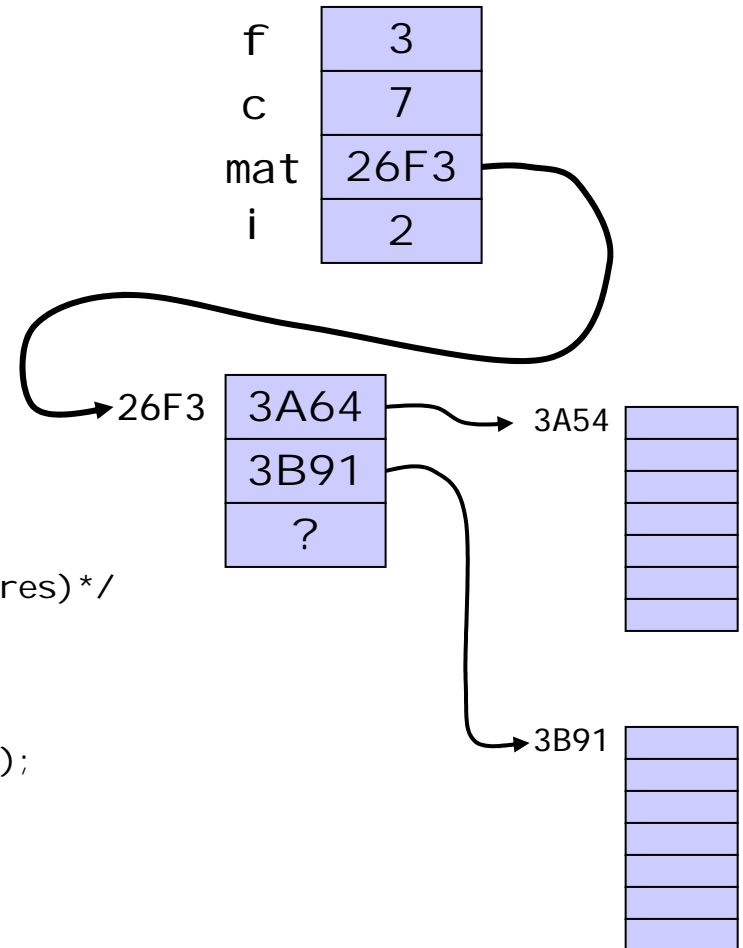
```
    mat=(double **)calloc(f,sizeof(double *));
```

```
    for(i=0; i<f; i++) {
```

```
        mat[i]=(double *)calloc(c,sizeof(double));
```

```
    }
```

```
    <... si gue... >
```



# Matrices dinámicas

```
/* Programa asignar una matriz dinámica */
```

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    int f; /* número de filas */
```

```
    int c; /* número de columnas */
```

```
    double **mat; /* puntero de puntero */
```

```
    int i; /* contador */
```

```
    /* obtengo f y c */
```

```
    f=3;
```

```
    c=7;
```

```
    /* Asigno memoria (falta el control de errores)*/
```

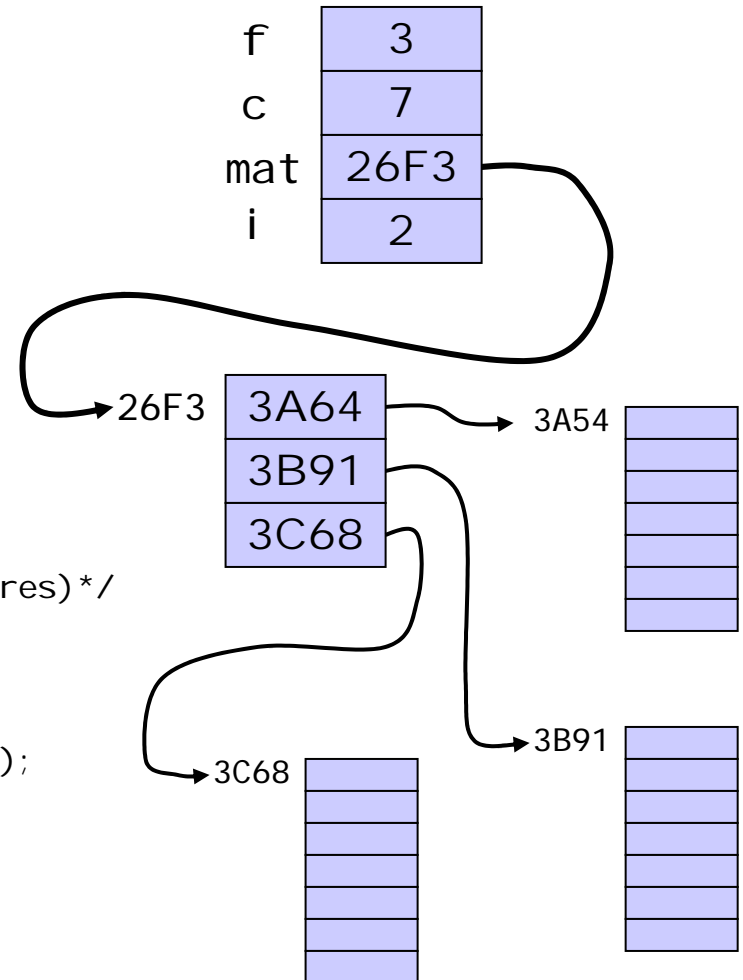
```
    mat=(double **)calloc(f,sizeof(double *));
```

```
    for(i=0; i<f; i++) {
```

```
        mat[i]=(double *)calloc(c,sizeof(double));
```

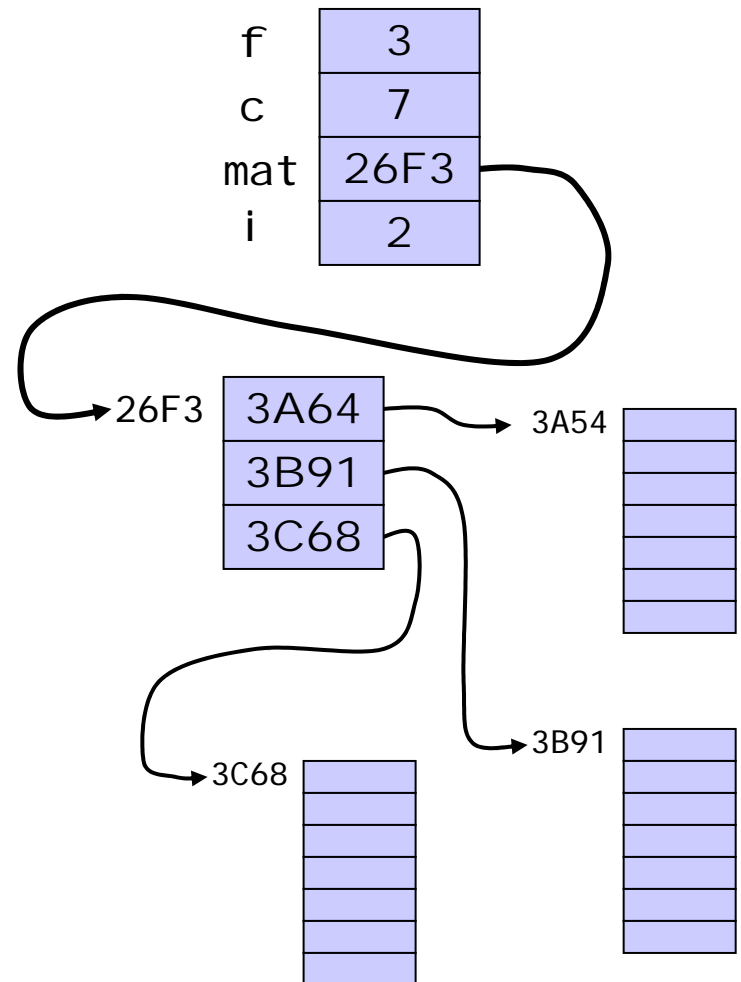
```
    }
```

```
    <... si gue... >
```



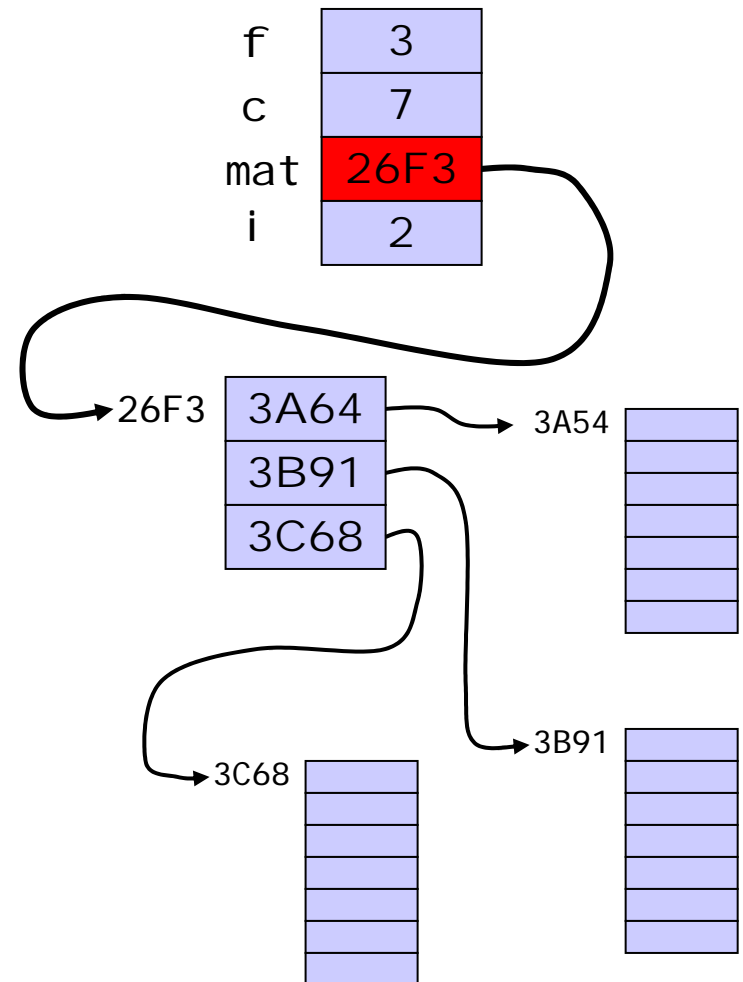
# Utilización

mat[2][5]=34.56;

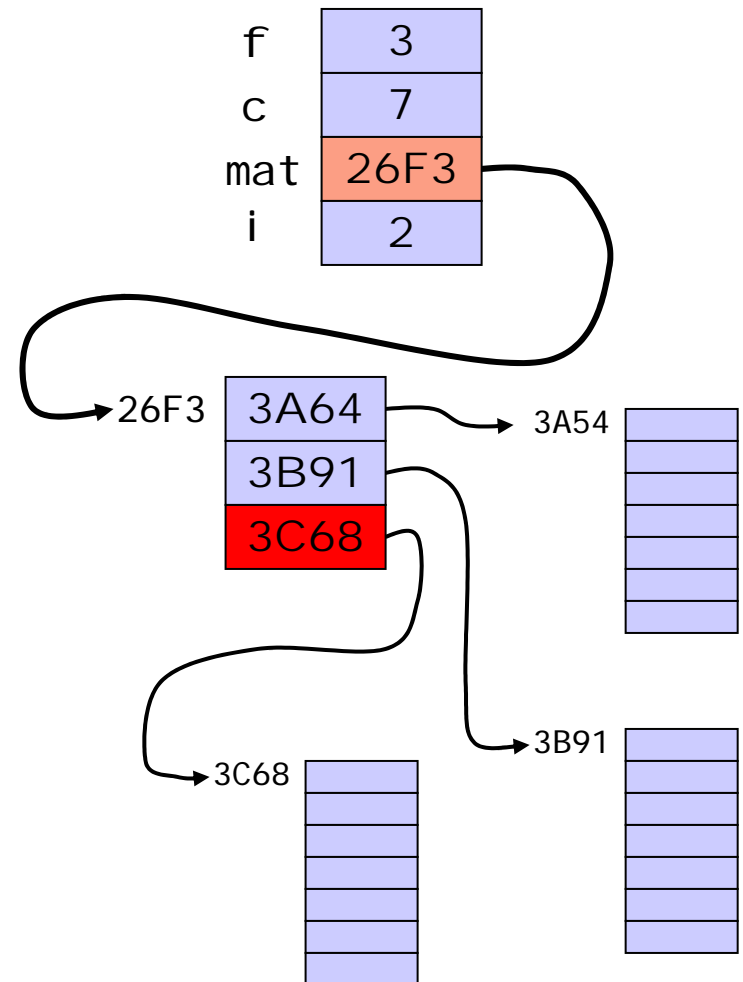
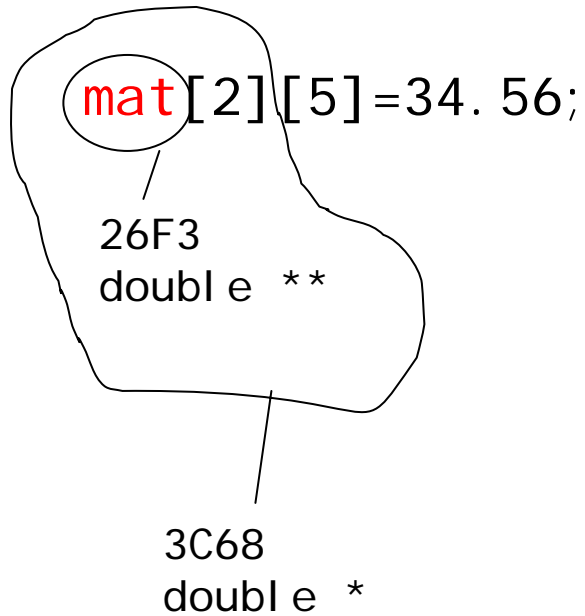


# Utilización

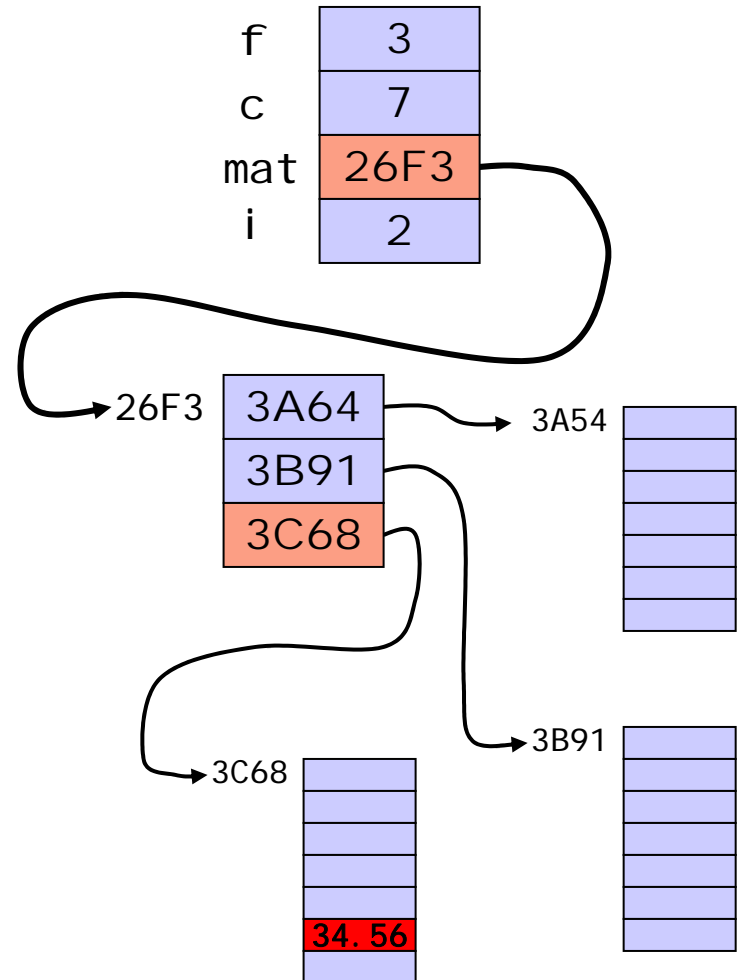
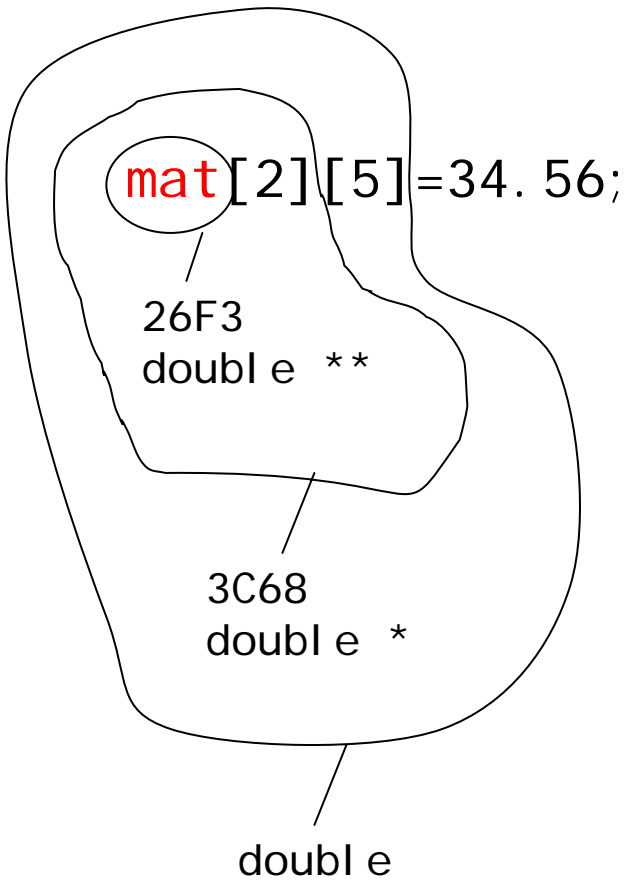
**mat**[2][5]=34.56;  
26F3  
double \*\*



# Utilización

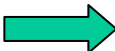


# Utilización

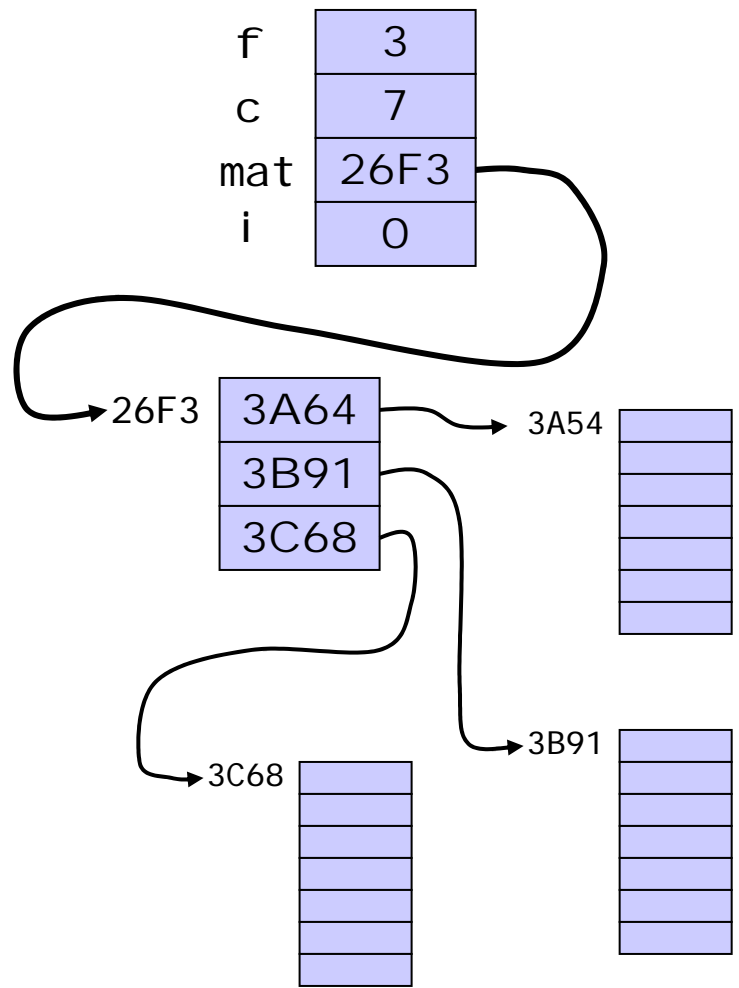




# Liberar Memoria



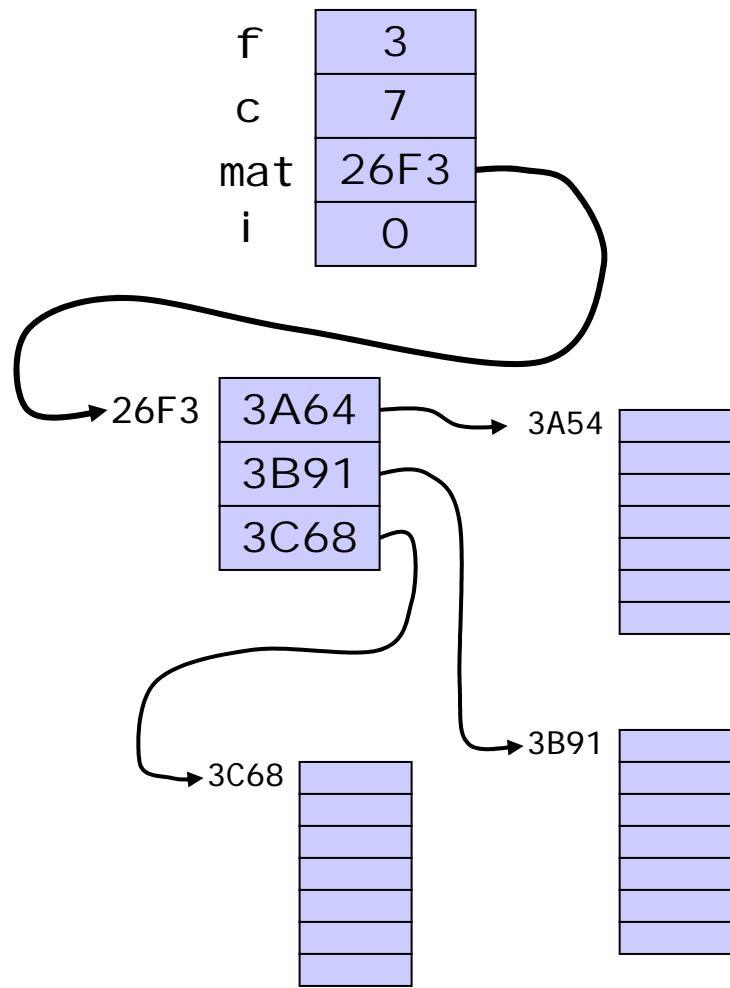
```
for(i=0; i<f; i++) {  
    free( mat[i] );  
}  
free(mat);
```



# Liberar Memoria

```
for(i=0; i<f; i++) {  
    free(mat[i]);  
}  
free(mat);
```

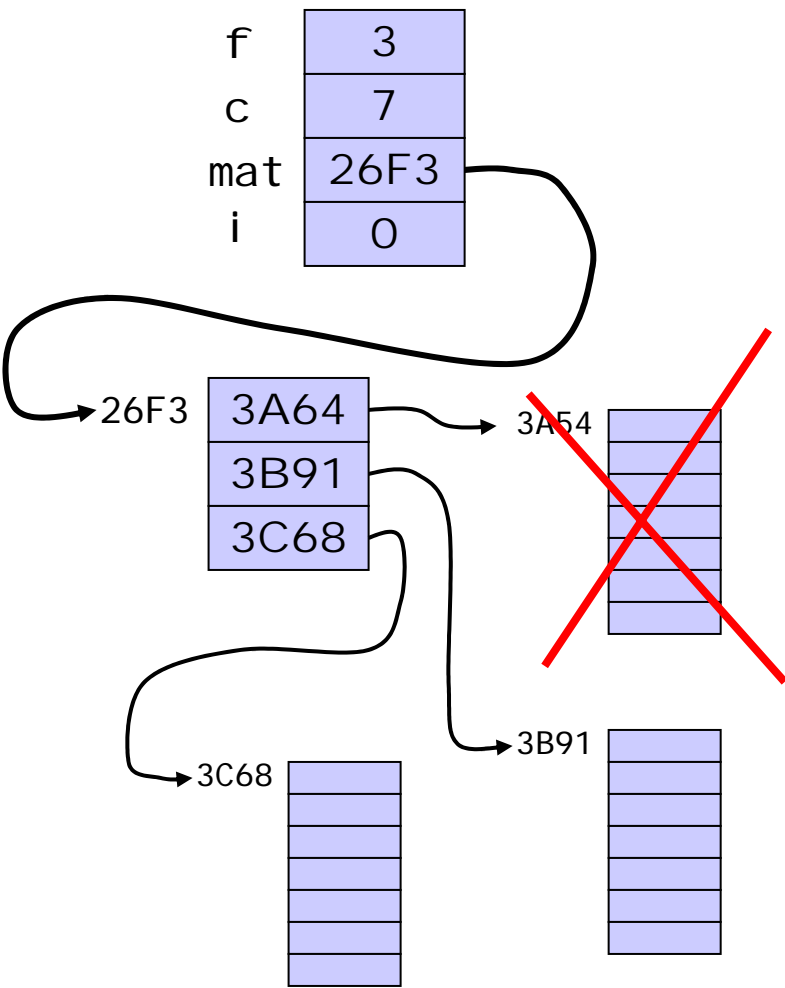
3A64



# Liberar Memoria

```
for(i=0; i<f; i++) {  
    free(mat[i]);  
}  
free(mat);
```

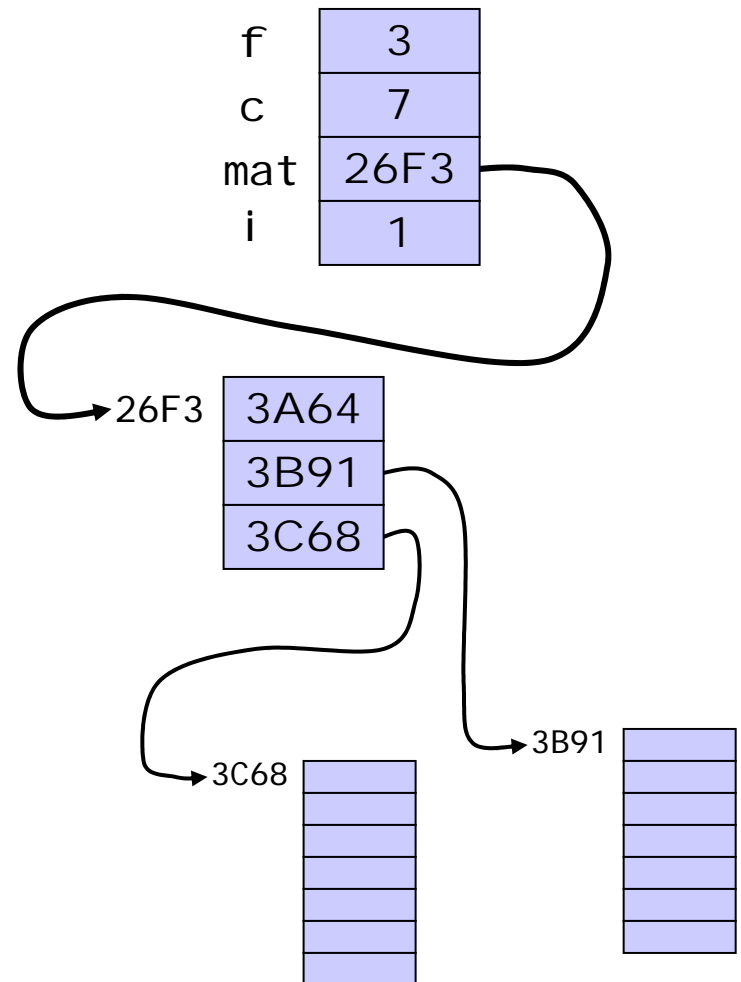
3A64



# Liberar Memoria

```
for(i=0; i<f; i++) {  
    free(mat[i]);  
}  
free(mat);
```

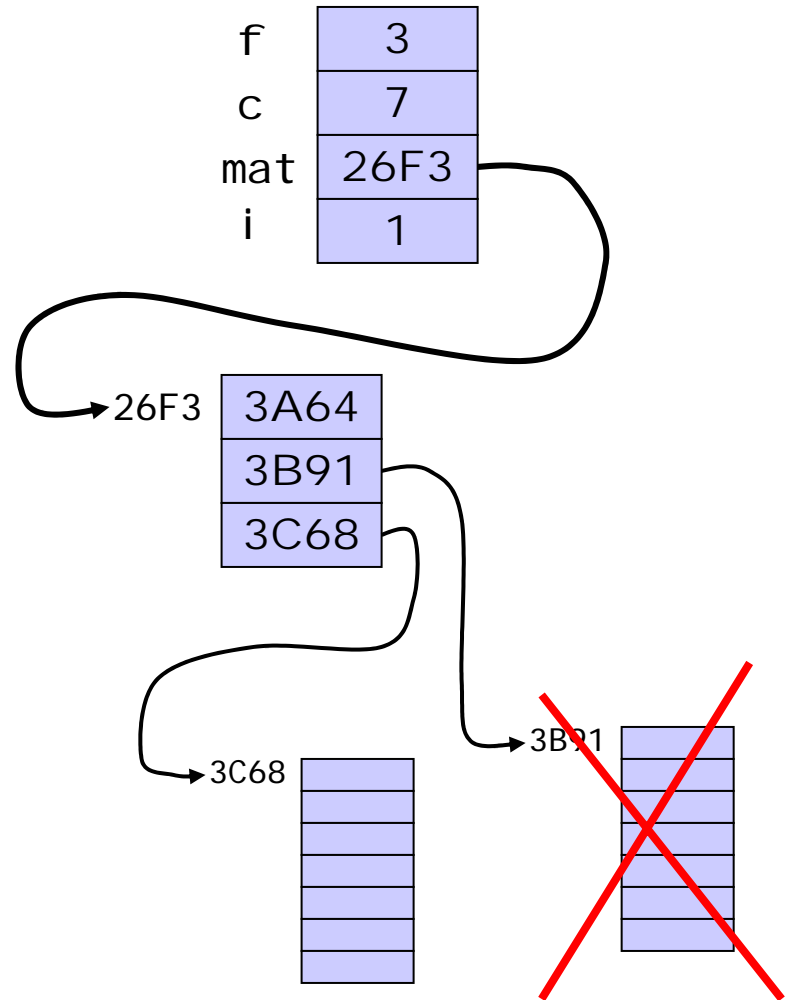
3B91



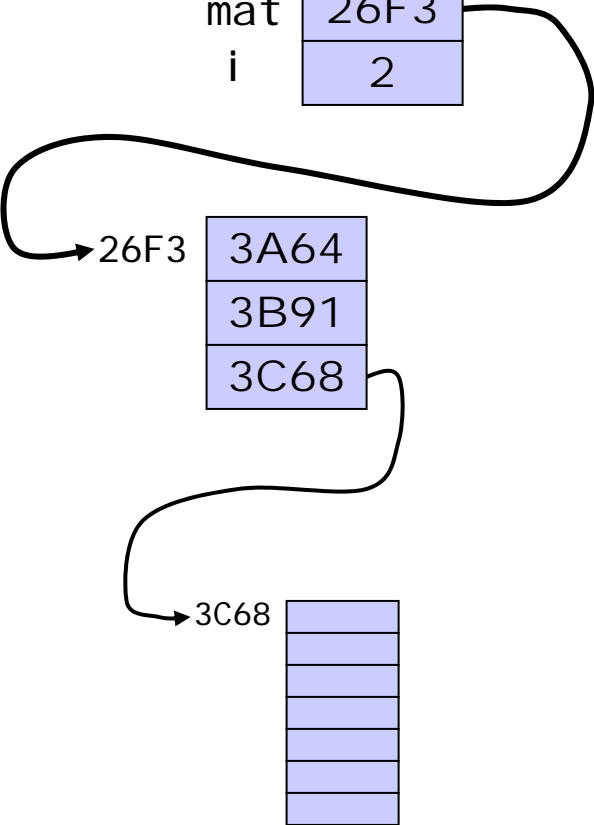
# Liberar Memoria

```
for(i=0; i<f; i++) {  
    free(mat[i]);  
}  
free(mat);
```

3B91



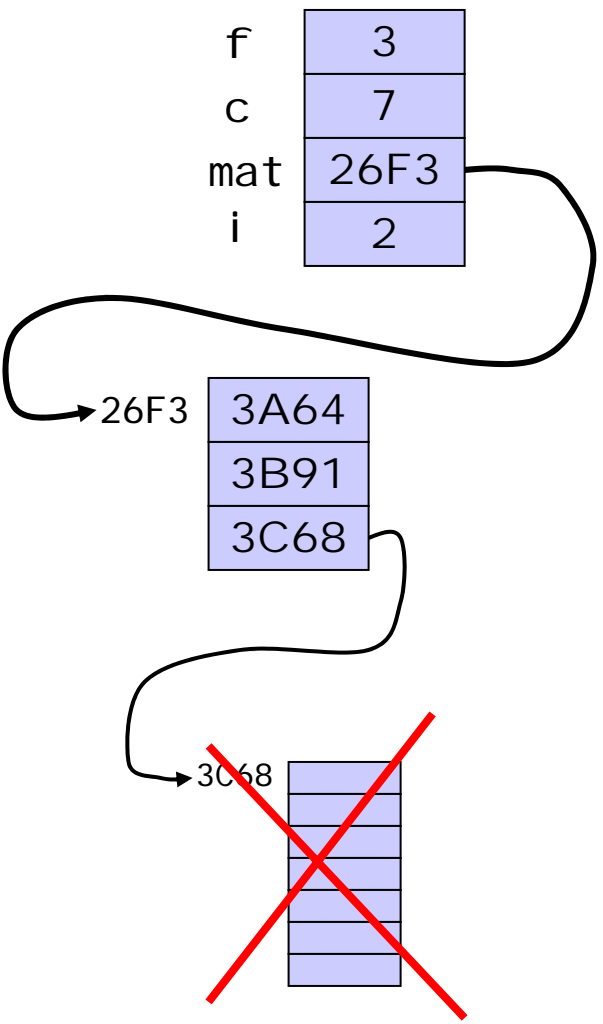
## Liberar Memoria



# Liberar Memoria

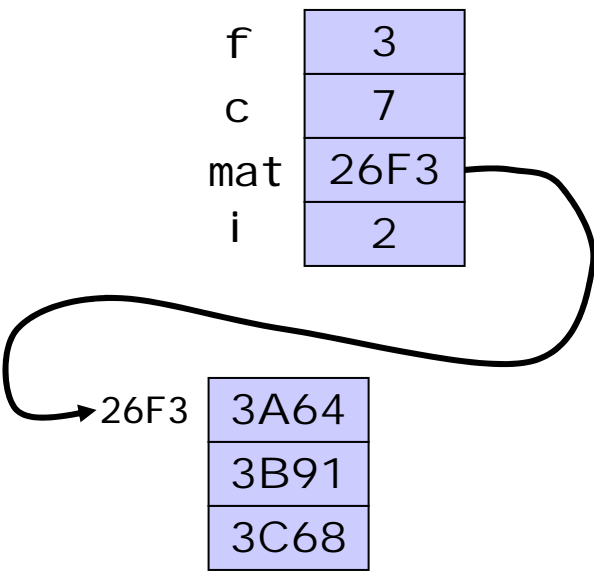
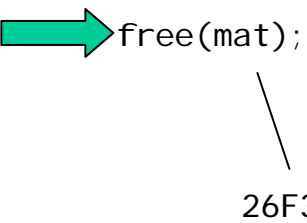
```
for(i=0; i<f; i++) {  
    free(mat[i]);  
}  
free(mat);
```

3C68



# Liberar Memoria

```
for(i=0; i<f; i++) {  
    free( mat[i] );  
}
```





# Liberar Memoria

```
for(i=0; i<f; i++) {  
    free( mat[i] );  
}  
free(mat);
```

26F3

f	3
c	7
mat	26F3
i	2

