

Estructura y Representación de Datos

Prof. Tatiana Ilabaca
Primer semestre 2021



Módulo 3
Estructuras de datos dinámicas

Punteros y Arreglos

Objetivos

Lección 7

- Conocer la relación entre punteros y arreglos
- Implementar arreglos de punteros
- Implementar arreglos de listas enlazadas

Arreglos & Punteros

Arreglo

- Un arreglo representa a un conjunto de celdas de **memoria consecutivas** (contiguas), bajo un mismo identificador (nombre)



int a[5]={1,2,3};

- El **identificador** de un arreglo representa a la dirección de memoria del primer elemento

a, dirección de memoria de la primera celda

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59
60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99

Arreglos & Punteros

Arreglo

- Vamos a comprobar:

```
#include <stdio.h>

void main()
{
    int a[5]={1,2,3};
    int* p;

    p=a;

    printf("&a: %p",&a);
    printf("\n a: %p",a);
    printf("\n p: %p",p);

}
```

1

2

3

Arreglos & Punteros

Arreglo

- Al código anterior agrega las siguientes líneas:

```
printf("\nPrimer elemento del arreglo: %i",a[0]);
```

```
printf("\nPrimer elemento del arreglo: %i",*a);
```

```
p=&a[1];
```

```
printf("\nSegundo elemento del arreglo: %i",a[1]);
```

```
printf("\nSegundo elemento del arreglo: %i",*p);
```

1

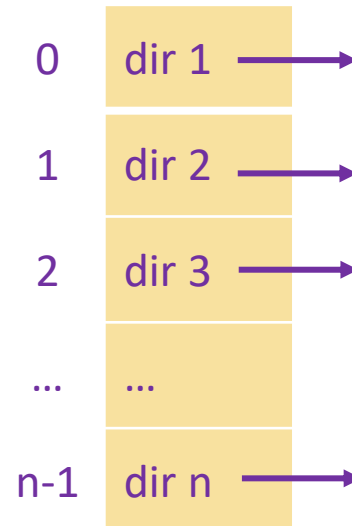
2

3

Arreglos & Punteros

Arreglo de punteros

- Los elementos del arreglo corresponden a punteros, es decir, es un arreglo de direcciones



```
#include <stdio.h>

#define N 5

void main()
{
    int* v[N];
}
```

```
#include <stdio.h>

#define N 5

typedef struct Nodo
{
    int dato;
    struct Nodo* sig;
}Nodo;

void main()
{
    Nodo* v[N];
}
```

Arreglos & Punteros

Arreglo de punteros

- Los elementos del arreglo corresponden a punteros, es decir, es un arreglo de direcciones

	0	1	...	m-1
0	dir	dir	...	dir
1	dir	dir	...	dir
2	dir	dir	...	dir
...
n-1	dir	dir	...	dir

```
#include <stdio.h>

#define N 5
#define M 7

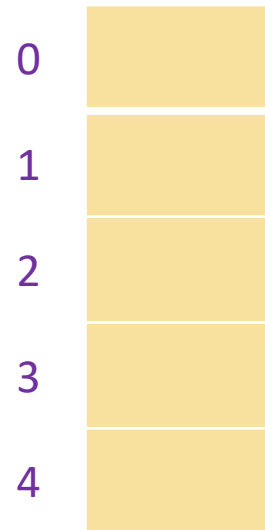
typedef struct Nodo
{
    int dato;
    struct Nodo* sig;
}Nodo;

void main()
{
    Nodo* m[N][M];
}
```

Arreglos & Punteros

Arreglo de punteros

- El acceso a los elementos del arreglo sería, por ejemplo:



```
#include <stdio.h>

#define N 5

typedef struct Nodo
{
    int dato;
    struct Nodo* sig;
}Nodo;

void main()
{
    Nodo* v[N];
    Nodo* aux;

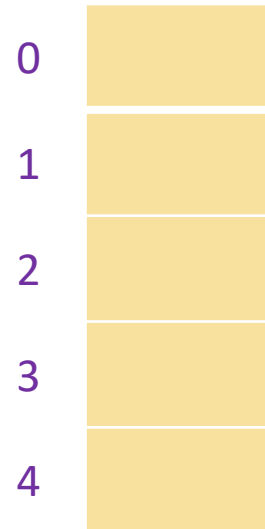
    v[0]=NULL;

    aux=(Nodo*)malloc(sizeof(Nodo));
    v[0]=aux;

    v[0]->dato=12;
    v[0]->sig=NULL;
}
```


Actividad 1

- Dada la estructura de nodo anterior, crea un arreglo de listas enlazadas simples
- El usuario debe determinar el tamaño de cada lista



Actividad 2

- Inserta un nuevo nodo en una lista indicada por el usuario