

9 de febrero del  
2018

- Horrieta Villegas  
Alfonso

## Ejercicio

### - Clasificación -

#### ► Evolución Histórica

- Lenguajes de 2da generación
- Lenguajes de 5<sup>o</sup> generación

#### ► Manera de ejecutarse

- Lenguajes interpretados
- Lenguajes compilados

#### ► Forma de abordar la

##### Tarea

- Lenguajes Imperativos
- Lenguajes Declarativos

#### ► Paradigma de programación

- Lenguajes Orientado a Objetos
- Lenguaje Funcionales

### - Grupo 1

(7:00 - 9:00)

### Ejercicio 1

10

## Ejercicios



- 14 de febrero de 2018
- Marrieta Villegas
- Alfonso
- Grupo 1 (7:00-9:00)
- Ejercicio 2

$$\begin{array}{r} \overline{28} \quad 8 \quad 21 \quad \overline{1234} \\ 1 | 10001011 | 0011 \dots \end{array}$$

$$\begin{array}{r} \text{negative} \\ - 139 \\ \hline 127 = E \end{array}$$

$$\left. \begin{array}{l} 2^{-3} = \frac{1}{8} \\ 2^{-4} = \frac{1}{16} \end{array} \right\} \cdot 1875 + 1$$

$$\therefore 2^{12} = \underline{4096}$$

$$\therefore (-1)(4096)(1.1875) = -4864 \quad //$$

b) 35,375

$$\begin{array}{r} 32168421 \\ \hline 100011 \end{array}$$

## Número Final

0 | 10000100 | 00011011  
↑      ↑  
Positive Exp

$$E + S ; 127 + 5 = 132$$

~~128 | 64 | 32 | 16 | 8 | 4 | 2 | 1~~ | 8 bits exponential  
~~1 | 0 | 0 | 0 | 0 | 1 | 0 | 0~~

$$\begin{array}{r} .375 \times 2 = .75 \\ .75 \times 2 = 1.5 \\ .5 \times 2 = 1.0 \end{array}$$

Número final | 0| 10000100|000110110000000000000000  
 ↓              ↑  
 Positivo    Exp

Es negativo = 1

c) -145.44

$$145 = \underline{128} \mid 64 \mid 32 \mid 16 \mid 8 \mid 4 \mid 2 \mid 1$$

1.0010001

7 saltos

$$E+7 = 134$$

$$134 = \underline{128} \mid 64 \mid 32 \mid 16 \mid 8 \mid 4 \mid 2 \mid 1 \rightarrow \text{Exp} \quad 110$$

$$.44 \times 2 = .88$$

$$.88 \times 2 = 1.76$$

$$.76 \times 2 = 1.52$$

$$.52 \times 2 = 1.04$$

$$.04 \times 2 = .08$$

$$.08 \times 2 = .16$$

... Lo mando !! perdon ...

Segundo envío

Nombre Murrieta Villegas Grupo: 1

- 1.- En las líneas de código que se indica escribe "C" si la afirmación es correcta o "I" si es incorrecta o generaría un error de compilación

```
int arreglo[8];
int var1 = 0
float b = 5.0;
int a = 2;
arreglo[3] = 1; C
arreglo[var1] = 4; C
arreglo[a++ = 2; C
arreglo[b] = 3 I
arreglo[a*4] = 6; C ← [a*4] ~ [8] ← se pone
arreglo[a*2] = 7; C
```

(a.2)

- 2.- Realiza la representación gráfica del arreglo resultante después de ejecutar las instrucciones siguientes:

```
int arreglo[4][3] = {2, 4, 6, 3, 6, 9, 8, 10, 12, 12, 15, 18} →
arreglo[1][2] = 110;
arreglo[0][1] = 30;
arreglo[3][3] = 45;
arreglo[2][1] = 199;
arreglo[0][2] = 300
```

0	2	30	300
1	3	6	110
2	8	199	12
3	12	15	45

Table final

- 3.- Indica la salida del siguiente código

```
void main() {
    int a[3][4][2] = {4, 8, 12, 16, 20, 24, ..., 76, 80, 84, 88, 92, 96};
    printf("\n Primer valor: %d \n", a[1][1][1]);
    printf("\n Segundo valor: %d \n", a[0][3][1]);
    printf("\n Tercer valor: %d \n", a[2][2][1]);
    printf("\n Cuarto valor: %d \n", a[2][0][0]);
    system("pause");
}
```

0	4	8
1	12	16
2	20	24
3	28	32

0	36	40
1	44	48
2	52	56
3	60	64

0	68	72
1	76	80
2	84	88
3	92	96

Salida

Primer valor: 48

Segundo valor: 32

Tercer valor: 88

Cuarto valor: 68

Nota: No di el salto de linea debido a \n

- Ejercicio - 4  
- 16 de febrero 2018  
- Murrieta Villegas Alfonso  
- Grupo 1 (7:00 - 9:00)

(8)

a) procedure mystery

```
a: integer;  
b: integer;  
  
procedure enigma(x,y)  
begin  
    y=y+b;  
    x=b+x;  
    b=x+b;  
    a=y;  
end enigma  
  
begin  
a=2 ; b=7;  
enigma (a,b);  
write (a); write(b);  
end mystery;
```

### SALIDAS

► Pasa por  
Valor  
a = 2  
b = 7 //

► Pasa por  
Referencia  
x = a  
y = b

y = y + b; y = 7 + 7; y = 14  
x = 7 + a; x = 7 + 2; x = 9  
b = 9 + 7; b = 16  
y = a = 14  
∴ a = 14  
b = 16 //

b) Indicar los valores que se piden

3: int a=10, b=20, c=30, d=100, r1, r2, r3;  
4: int \*p1, \*p2, \*p3, \*p4;  
p1=&a; p2=&b; p3=&c; p4=p1;  
r1=\*p1+\*p2; r1=10+20; r1=30  
r2=\*p3-\*p4; r2=30\*10; r2=300  
r3=\*p2-\*p4; r3=20\*10; r3=200  
\*p1=44;  
\*p2=23;  
\*p3=d;

d=44 //  
d=23 //  
c=100 //  
r1=30 //  
r2=300 //  
r3=200 //

- (T3)
- Ejercicio 5 -
  - 28 de febrero de 2018
  - Muriel Villegas Alfonso
  - Grupo 1 (7:00 - 9:00):

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

```
struct direccion {
```

```
    char calle[40];
```

```
    int numero;
```

```
    char colonia[40];
```

```
    int cp;
```

```
} typedef struct direccion Direccion;
```

```
struct alumno {
```

```
    int numCuenta;
```

```
    char nombre[20];
```

```
    char apellido[20];
```

```
    float promedio;
```

```
    struct Direccion;
```

Identificador

```
} typedef struct alumno Alumno;
```

```
main() {
```

```
    Alumno alumno1, *ap; ap = &alumno1;
```

```
    alumno1.numCuenta = 31504;
```

```
    alumno1.nombre = "Alfonso";
```

```
    alumno1.apellido = "Muriel";
```

```
    alumno1.direccion.char = "calle...";
```

```
    alumno1.direccion.numero = "85717781";
```

```
    alumno1.direccion.colonia = "colonía";
```

```
    alumno1.direccion.cp = "11031";
```

```
    ap->direccion.calle = "Calle 2";
```

```
}
```

- Ejercicio 6
- 21 de Marzo de 2018
- Horario Villegas Alfonso
- Grupo 1 (7:00 - 9:00)

$$I] \quad ((12+8) \times 5) \div ((20 \times 2) - (7 \times 5))$$

$$\begin{array}{c}
 \begin{array}{ccccc}
 \begin{array}{|c|c|} \hline 8 & + \\ \hline 12 & \end{array} & \Rightarrow & \begin{array}{|c|c|} \hline 9 & \times \\ \hline 20 & \end{array} & \Rightarrow & \begin{array}{|c|c|} \hline 2 & \div \\ \hline 20 & \end{array} \\
 & & \begin{array}{|c|c|} \hline & \div \\ \hline & \end{array} & & \begin{array}{|c|c|} \hline x & \div \\ \hline & \end{array} \\
 & & & & \Rightarrow \begin{array}{|c|c|} \hline 40 & \div \\ \hline 40 & \end{array} \\
 & & & & \begin{array}{|c|c|} \hline 1 & \div \\ \hline 1 & \end{array} \\
 & & & & \Rightarrow \begin{array}{|c|c|} \hline & \div \\ \hline & \end{array}
 \end{array} \\
 \xrightarrow{\quad} \begin{array}{ccccc}
 \begin{array}{|c|c|} \hline 5 & \times \\ \hline 7 & - \\ \hline 40 & \div \\ \hline 100 & \end{array} & \Rightarrow & \begin{array}{|c|c|} \hline 35 & \div \\ \hline 40 & \end{array} & \Rightarrow & \begin{array}{|c|c|} \hline 5 & \div \\ \hline 100 & \end{array} \\
 & & \begin{array}{|c|c|} \hline & \div \\ \hline & \end{array} & & \begin{array}{|c|c|} \hline & \div \\ \hline & \end{array} \\
 & & & & \Rightarrow \begin{array}{|c|c|} \hline 20 & \div \\ \hline & \end{array} \\
 & & & & \begin{array}{|c|c|} \hline & \div \\ \hline & \end{array}
 \end{array}
 \end{array}$$

$$\therefore R.F = \frac{20}{f}$$

$$\text{II}] \left( (90+5) + (20 \div 2) \right) \times 5$$

$$\begin{array}{c}
 \begin{array}{|c|c|} \hline 5 & \\ \hline 90 & + \\ \hline \end{array} \Rightarrow \begin{array}{|c|c|} \hline 2 & \\ \hline 20 & \div \\ \hline 45 & + \\ \hline \end{array} \rightarrow \begin{array}{|c|c|} \hline 10 & \\ \hline 95 & + \\ \hline \end{array} \rightarrow \begin{array}{|c|c|} \hline 5 & \\ \hline 105 & x \\ \hline \end{array} \rightarrow \\
 \rightarrow \begin{array}{|c|c|} \hline 525 & \\ \hline & + \\ \hline \end{array} \quad \therefore R.F = \underline{\underline{525}} //
 \end{array}$$

III] 40, 5, +, 20, 8, 3, ÷, -, +

$$\begin{array}{c}
 \left| \begin{array}{r} 5 \\ 40 \end{array} \right| \xrightarrow{+} \left| \begin{array}{r} 3 \\ 8 \\ 20 \\ 45 \end{array} \right| \xrightarrow{\div} \left| \begin{array}{r} 8/3 \\ 20 \\ 45 \end{array} \right| \xrightarrow{-} \left| \begin{array}{r} 5/2 \\ 40 \end{array} \right| \xrightarrow{+} \left| \begin{array}{r} 87 \\ 3 \end{array} \right|
 \end{array}$$

$$\boxed{\frac{187}{3}} = \boxed{62.33} \quad \therefore R.F. = \frac{187}{3} \approx 62.333$$

7.5

Ejercicio 7  
 - 4 de Abril de 2018  
 - Morrieta Villegas Alfonso  
 - Grupo 1

- a) Escribir el código de búsqueda en una lista ligada simple
- b) Escribir el código para insertar un nodo en una lista ligada simple en la posición 3

a) Nodo tmp;  
~~tmp = head;~~  
~~while (tmp.next != 10)~~  
~~tmp = tmp.next~~

//Falta XD

b) Nodo nodo3;  
~~nodo3.info = 10;~~  
 Nodo tmp;  
~~tmp.info = 6; tmp.next = head;~~  
~~head.next = head; ?~~  
~~head.next.next = nodo3;~~

sirve para desplazar todos una posición y en el tercero que es  $\Rightarrow$   
 $head.next.next$  le insertamos directamente el nodo 3 (creado).

③

► Murrieta Villegas Alfonso

► EDA 1

► Ejercicio 8

4) ► Código para invertir los elementos de una lista simple. {caso de 3 elementos}

// se crea lista



nodo + tail;

tail.next = head.next

head.next = head

head.next = null

// están invertidos  
falta generalizar ?

Para general  
Meter en ciclo

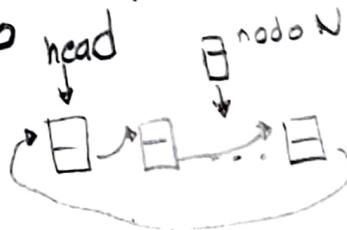
► insertar un elemento en la posición n de una lista circular

int size; // definir tamaño head  
nodo head; nodo N

nodo nodoN;

nodoN.info = ?;

posición = n; // aquí el n es  
cualquier número



if ( n != size )

nodoN.next = head.next.next

head.next.next = nodoN

cantidad desplazados

?

Meter en  
ciclo

else

print ( Ingrese una posición dentro de la  
cantidad de elementos )

// otra alternativa para comparar es que dentro de los  
ciclos se considere el size como límite de la lista

- Munietka Villegas Alfonso
- EDA 1
- Edgar Tista
- Ejercicio 9

7

Escribir el pseudocódigo para

- eliminar el nodo 3
- Agregar un nuevo nodo entre 4 y 5



~~/\* num es la posición a borrar \*/~~

Nodo tmp = head;  
tmp = list.next.next;

tmp.next = list.prev.prev X  
tmp.prev = list.next.next  
borrar I esimo (Nodo 3)

list.next = tmp.prev.  
list.prev.next = tmp.next

Nodo Nuevo = head;

: Pendiente ?



b (En caso correcto)

# Problema de la mochila

(a3)

## Ejercicio 10

Marieta Villegas Alfonso

- Grupo 1
- EDA-I

DATOS	Peso Costo	10	20	30	45	50	60	70	mayor valor
		30	35	55	75	65	80	60	
		3	1,75	1,83	1	1,3	1,333	1,333	1,867142
	Proporción de costo peso								
	• Capacidad = 150 [kg]								

### 1] Mochila Normal

#### ► Primer Caso (Más ligero)

$$\text{Peso} = 10 + 20 + 30 + 45 + 80 = 155 \text{ [kg]} \quad \checkmark$$

$$\text{Valor} = 30 + 35 + 55 + 45 = \$165 \quad \cancel{\checkmark}$$

#### ► Segundo caso (El más caro)

$$\text{Peso} = 60 + 50 + 30 = 140 \text{ [kg]} \quad \text{aun cabe el de 10 kg}$$

$$\text{Valor} = 80 + 65 + 55 = \$200$$

#### ► Tercer caso

$$\text{Mayor Valor} = 3 \times 1,83, 1,75, 1,333 \rightarrow \text{ya no caben}$$

$$\text{Peso} = 10 + 30 + 20 + 60 = 120 \text{ [kg]}$$

$$\text{Valor} = 30 + 55 + 35 + 80 = \$200 \quad \checkmark$$

### 2] Mochila Fraccional

$$\begin{array}{l} 30 = 65 \\ 45 = 58,5 \end{array}$$

#### ► Primer caso (Más ligero)

$$\text{Peso} = 10 + 20 + 30 + 45 + 45 (\text{de } 50) = 150 \text{ [kg]}$$

$$\text{Valor} = 30 + 35 + 55 + 45 + 45 = 223,5 \text{ [kg]} \quad \cancel{\checkmark}$$

#### ► Segundo caso (Más caro)

$$\text{Peso} = 60 + 50 + 40 (\text{de } 70) = 150 \text{ [kg]} \quad \begin{array}{l} 70 = 60 \\ 40 = \end{array}$$

$$\text{Valor} = 80 + 65 + 39,28 = 179,28 \text{ \$}$$

#### ► Tercer caso

$$\begin{array}{l} 50 = 65 \\ 30 = 39 \end{array}$$

$$\text{Peso} = 10 + 30 + 20 + 60 + 30 (\text{de } 50) = 150 \text{ [kg]}$$

$$\text{Valor} = 30 + 55 + 35 + 80 + 39 = \$239 \quad \checkmark$$

• Morrieta Villegas Alfonso

• Grupo 1

• E D A - I

Actividad 11  
(Ejercicio)

9

## Funciones recursivas

- ① Imprimir números impares entre 0 y n de manera descendente
- ② Que determine el tamaño de una lista ligada simple

① int sumaImpares (int n) {

    if (n == 1)  
        return 1;

    else

        return sumaImpares (n+1) + n

}

► caso base n == igual

1/af

1/af es el caso base de los impares

(n+1)

Parte  
recursiva

② int TamañoLista (Lista lista1) {

    if (tmp.next == null) {

        return 0; tamaño = 0

} else

    tamaño = (tamaño + 1) +

    TamañoLista (tmp.next);

    tmp = tmp.next;

    Añadir primero (Lista lista1)

    return (Lista lista1)

}

El caso base es el  
ultimo elemento?

Murrieta Villegas Alfonso  
• (Actividad) 12  
Ejercicio

• EDA 1

• Grupo 7:00 - 9:00 a.m 1

9

- 1)  $\pi \rightarrow O(1)$  Constante
- 2)  $\log(\log(n)) \rightarrow$  Logarítmico
- 3)  $\log(n) \rightarrow$  Logarítmico
- 4)  $\sqrt[6]{n} = n^{1/6} \rightarrow$  Polinomial X
- 5)  $\sqrt{n} = n^{1/2} \rightarrow$  Polinomial X
- 6)  $(\log(n))^4 = (\log(n))^{\text{contar}} \rightarrow$  Lineal-logarítmico / Logarítmico
- 7)  $1714n \rightarrow$  Lineal
- 8)  $n^2 \rightarrow n^c$  Polinomial Cuadrática
- 9)  $100n^3 \rightarrow n^c$  Polinomial Cúbica
- 10)  $10n^4 \rightarrow n^c$  Polinomial 4º grado
- 11)  $(\frac{3}{2})^n \rightarrow c^n$  Exponencial
- 12)  $2^n \rightarrow c^n$  Exponencial
- 13)  $8n! \rightarrow n!$  Factorial

ordenadas

- |   |                      |                       |
|---|----------------------|-----------------------|
| ① | 1) $\pi$             | 11) $(\frac{3}{2})^n$ |
| ② | 2) $\log(\log(n))$   | 12) $2^n$             |
| ③ | 3) $\log(n)$         | 13) $8n!$             |
| ④ | 4) $\log(\log(n))^4$ |                       |
| ⑤ | 5) $1714(n)$         |                       |
| ⑥ | 6) $\sqrt[6]{n}$     |                       |
| ⑦ | 7) $\sqrt{n}$        |                       |
| ⑧ | 8) $n^2$             |                       |
| ⑨ | 9) $100n^3$          |                       |
| ⑩ | 10) $10n^4$          |                       |

Grupo: 1

EDA: 1

Murieta Villegas Alfonso

Edgar Tista

### Ejercicio 13

(9)

① ~~funcion(i)~~

```
for(ent l=0; i=1; i++;; i++)  
    For(
```

función 1 () {  
 for(~~cnt+l=0; l=1; l<=n; l++~~;  
 for(~~j=1; j<n; j++~~;  
 ~~l++;~~)  
 }  
  $\therefore O(n^2)$

Polinomio  
Cuadrático

② ~~funcion2(l)~~

```
for(cnt=0, l=1; l<=n; l*=2)  
    for(j=1; j<=n; j++)  
        cnt+=t; int a=0; int b=10;
```

complejidad

~~$O(n^2)$~~   
 $\therefore 3 \cancel{O(n^2)}$

Polinomio  
cuadrático

③ ~~funcion3(int n)~~

```
if(n <= 0)  
    return 1;
```

```
else  
    return 1 + funcion3(n-8);
```

complejidad

~~$O(n)$~~

Línea

$\therefore \Omega \rightarrow O(n)$  Línea  
 $\therefore \Theta = O(n)$

① Argumento es  $n^2$  debido a que tiene dos ciclos For anidados  
de 1 a n o sea un n para cada n

② Pese esta el " $c^{*}=2$ " de igual forma seguiría siendo

③ Solo se llama una vez la función recursiva.