

## ESTRUCTURA DE DATOS SOLEMNE III

- 1) [20P] Dado dos conjuntos A, B, donde A corresponde al conjunto de asignaturas que deben inscribir el próximo semestre Pedro de la carrera Ingeniería en Computación e Informática, y B es el conjunto de asignaturas que debe inscribir Pablo de la misma carrera. Considere que Pedro inscribe las siguientes asignaturas: Bases de Datos, Programación II, inglés, Autómatas; y Pablo debe inscribir: Cálculo diferencial, Bases de Datos, inglés, Sistemas Operativos, Química. Considere, además, que los elementos del Universo si pueden numerar de la siguiente forma:

0: inglés                      1: Programación II                      2: Autómatas                      3: Bases de Datos  
4: Sistemas Operativos                      5: Cálculo Diferencial                      6: Química                      7: Diseño de SW

Usted debe representar los conjuntos A, B, mediante una representación de conjuntos de **bits** y responda cada pregunta justificando muy bien a través de la realización de las operaciones necesarias paso a paso:

Representación bits:

	7	6	5	4	3	2	1	0
A:	0	0	0	0	1	1	1	1
B:	0	1	1	1	1	0	0	1

- a) ¿Comparten alguna asignatura ambos estudiantes y si es así cuáles son?.

$A \cap B = \{00001001\}$  Comparten la asignatura 3 y 0, o sea, Bases de Datos y Inglés.

- b) El conjunto de asignaturas a inscribir por Pedro ¿es un subconjunto del conjunto de asignaturas de Pablo?.

¿ $A \subset B$ ?  $\Rightarrow$  A es un subconjunto de B cuando  $A \cap B = A$

$A = 00001111$   
 $A \cap B = 00001001$   
 $A \cap B \neq A \therefore A$  no es un subconjunto de B.

- c) ¿Pablo tiene contemplado inscribir la asignatura Diseño de SW?.

¿ $7 \in A \cap B$ ?  $x \% 2 = x \& 1$

$B \gg 7 \Rightarrow B: 01111001$   
 $B \gg 7: 00000000$

$B \gg 7 \% 2 = B \gg 7 \& 1$

$B \gg 7: 00000000$   
 $\% 2: 00000001$   
 $\therefore 00000000 \Rightarrow 0 //$

$\therefore 7 \notin B$

- d) Lamentablemente Pablo reprobó la asignatura Estructuras de Datos, por lo tanto, no podrá inscribir Bases de Datos pero si podrá inscribir Autómatas (eliminar y después insertar). ¿Cómo quedaría finalmente el conjunto de asignaturas que debe inscribir Pablo?

$$\begin{array}{r}
 \text{A) Eliminar Bases de Datos del conjunto B. } \Rightarrow B - (1 \ll 3): \\
 B: 0 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \\
 -(1 \ll 3): 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \\
 \hline
 \therefore A - (1 \ll 3) = 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 1 \\
 \text{se eliminó B.D.}
 \end{array}$$

$$\begin{array}{r}
 \text{ii) } 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 1 \\
 + 1 \ll 2: 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \\
 \hline
 0 \ 1 \ 1 \ 1 \ 0 \ 1 \ 0 \ 1 \\
 \text{se agregó AUTÓMATAS}
 \end{array}$$

- 2) [20P] Implemente la pregunta 1 utilizando lenguaje C (implementación binaria). Debe implementar todo lo necesario (funciones y main) para responder cada pregunta del enunciado. Las operaciones deben ser implementadas a través de funciones y el main debe crear los conjuntos, llamar a las funciones y mostrar los resultados de cada pregunta.

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

```
void show_set(unsigned int C);
```

```
unsigned int Interseccion(unsigned int A, unsigned int B);
```

```
unsigned int pertenecer(unsigned int C, int x);
```

```
unsigned int Add(unsigned int C, int x);
```

```
unsigned int Delete(unsigned int C, int x);
```

```
unsigned int esSubConjunto(unsigned int A, unsigned int B);
```

```
int main() {
```

```
    unsigned char A, B;
```

```
    A = B = 0;
```

```
    A = Add(A, 0);
```

```
    A = Add(A, 1);
```

```
    A = Add(A, 2);
```

```
    A = Add(A, 3);
```

```
    B = Add(B, 0);
```

```
    B = Add(B, 3);
```

```
    B = Add(B, 4);
```

```
    B = Add(B, 5);
```

```
    B = Add(B, 6);
```

```
    printf("A (asignaturas a inscribir por Pedro): "); show_set(A);
```

```
    printf("B (asignaturas a inscribir por Pablo): "); show_set(B);
```

```
//RESPONDER a): CUÁLES SON LAS ASIGNATURAS QUE COMPARTEN PEDRO y PABLO
```

```
    printf("\nLas asignaturas compartidas por Pedro y Pablo son (A & B): ");
```

```
    show_set(Interseccion(A, B));
```

```
//RESPONDER b): A ES SUBCONJUNTO DE C
```

```
    if ( esSubConjunto(A,B)==1){
```

```
        printf("\nConjunto A SI es un subconjunto de B ");
```

```
    }else{
```

```
        printf("\nConjunto A NO es un subconjunto de B ");
```

```
    }
```

```
//RESPONDER c): 7 (Diseño de SW) pertenece al conjunto B: asignaturas a inscribir por Pablo
```

```
    if (pertenecer(B,7)==1){
```

```
        printf("\n\nAsignatura 7 (Diseño de SW) SI pertenece al conjunto B");
```

```
    }else{
```

```
        printf("\n\nAsignatura 7 (Diseño de SW) NO pertenece al conjunto B");
```

```
    }
```

```
//RESPONDER d): Eliminar Base de Datos e Insertar Autómatas del conjunto B
```

```

Delete(A, 3);
printf("\n\nDespues de ELIMINAR asignatura 3 (Base de Datos) e INSERTAR asignatura 2 (Autómatas): ");
show_set(Add(B, 2));
return 0;
}

```

## //FUNCIONES

```

void show_set(unsigned int C) { //función que muestra los elementos de un conjunto binario
    int i;
    for (i = 0; i < 8; i++) {
        printf("%d", !!(C << i) & 0x80));
    }
    printf("\n");
}

```

```

unsigned int Interseccion(unsigned int A, unsigned int B) { //función que hace la intersección de dos conjuntos
    return A & B;
}

```

```

unsigned int esSubConjunto(unsigned int A, unsigned int B) { //Revisa si un conjunto A es subconjunto de B
    return Interseccion(A, B) == A;
}

```

```

unsigned int pertenecer(unsigned int C, int x) { //función que revisa si un elemento pertenece a un conjunto
    return (C >> x)%2;
}

```

```

unsigned int Add(unsigned int C, int x) { //función que agrega elementos a un conjunto
    return C + (1 << x);
}

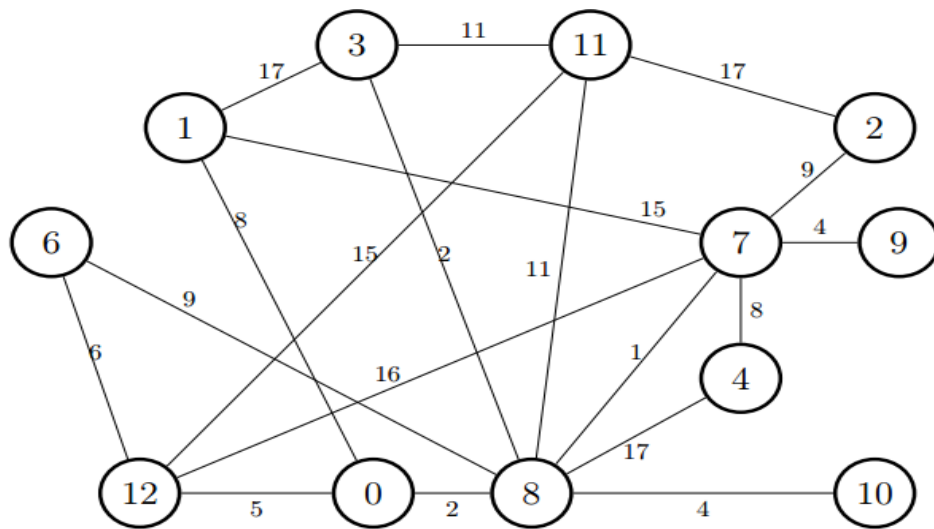
```

```

unsigned int Delete(unsigned int C, int x) { //función que elimina un elemento en un conjunto
    return C - (1 << x);
}

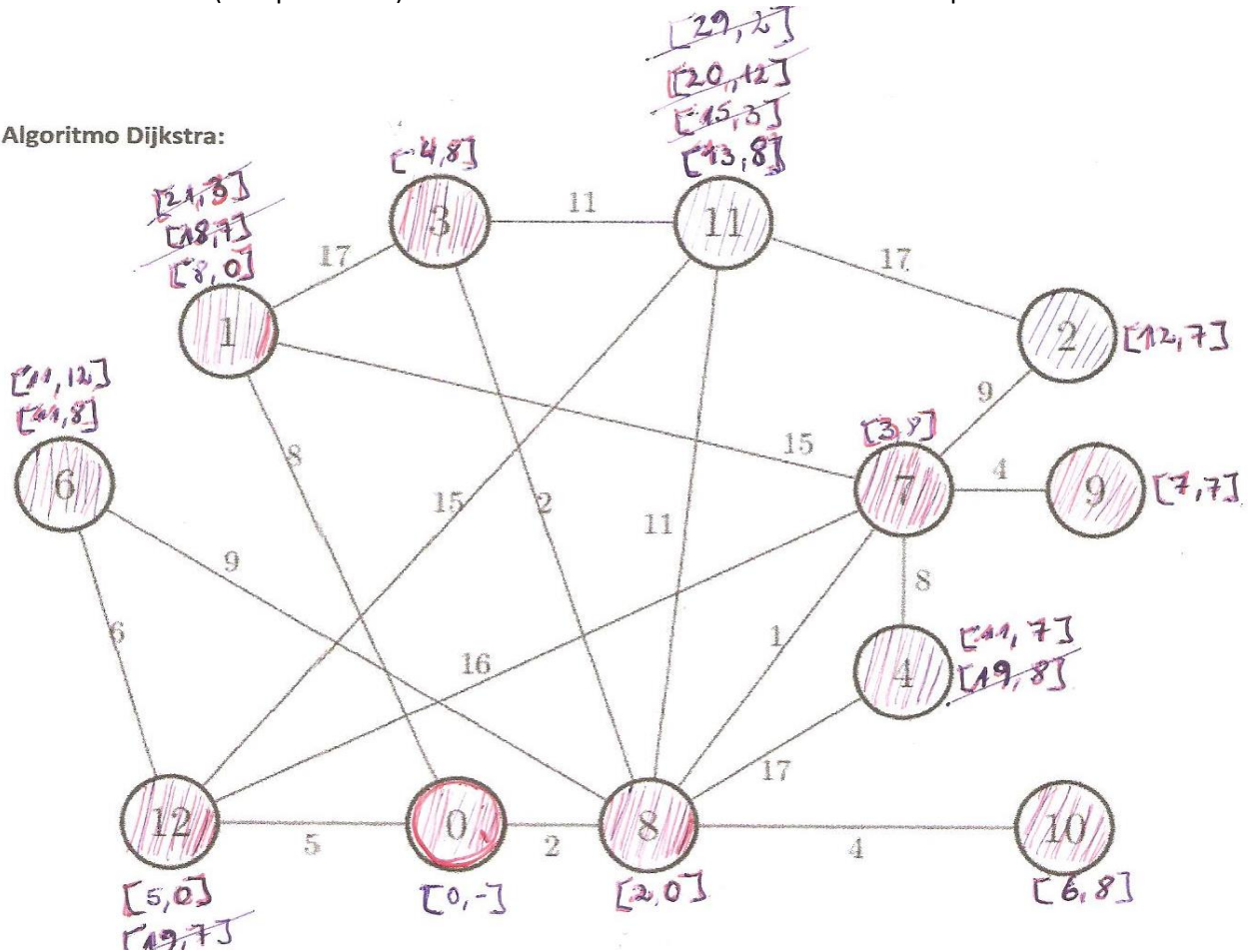
```

- 3) Para el grafo siguiente (considere origen nodo 0 y criterio recorrido el mínimo costo, subcriterio orden de nodos):



- a) [20P] Utilice el algoritmo de Dijkstra para determinar el camino más corto y el costo entre el nodo 0 y el nodo 11, y entre el nodo 0 y el 6. Además, de todos los recorridos que entrega el algoritmo, ¿cuál de todos ellos es el más económico? (indique la ruta). Fundamente claramente cada una de sus respuestas.

Algoritmo Dijkstra:



\* Camino más corto entre 0 y 11:

Recorrido: 0 - 8 - 11 y costo 13 (3)

\* Camino más corto entre 0 y 6:

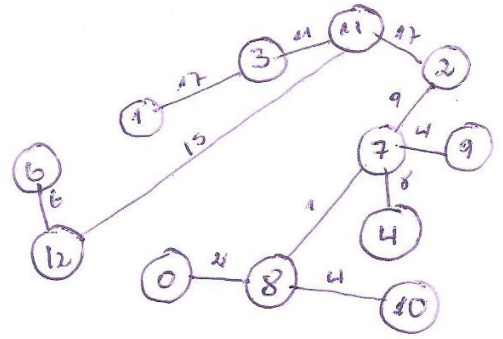
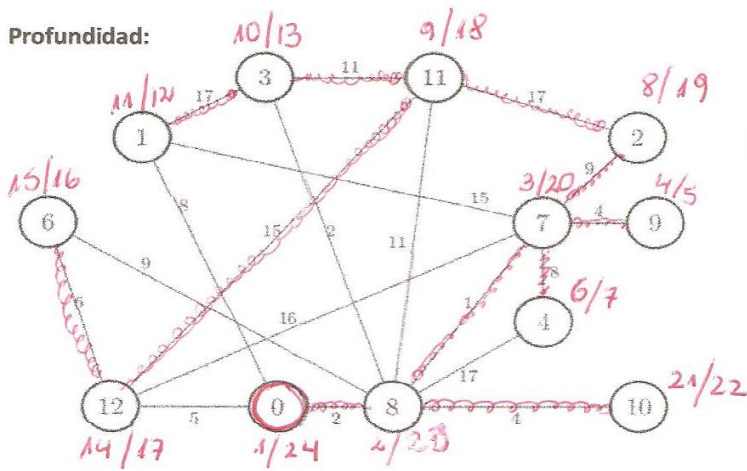
Recorrido: 0 - 8 - 6 y costo 11 (2)

Recorrido: 0 - 12 - 6 y costo 11 (2)

\* El recorrido más económico desde el origen es el entre el nodo 0 y el 8: 0 - 8 y costo 2 (3)

- b) [30P] Realice paso a paso el recorrido de Búsqueda por Anchura y por Profundidad. Para cada caso, indique claramente los diferentes arcos para construir el Bosque Abarcador por Anchura. Dibuje el grafo final y señale expresamente su salida.

Profundidad:

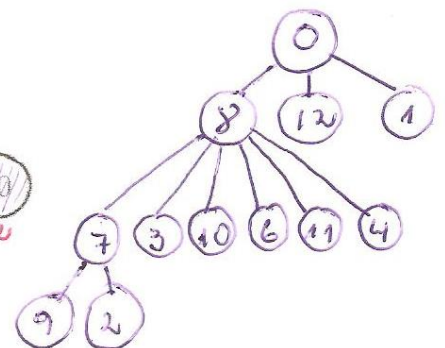
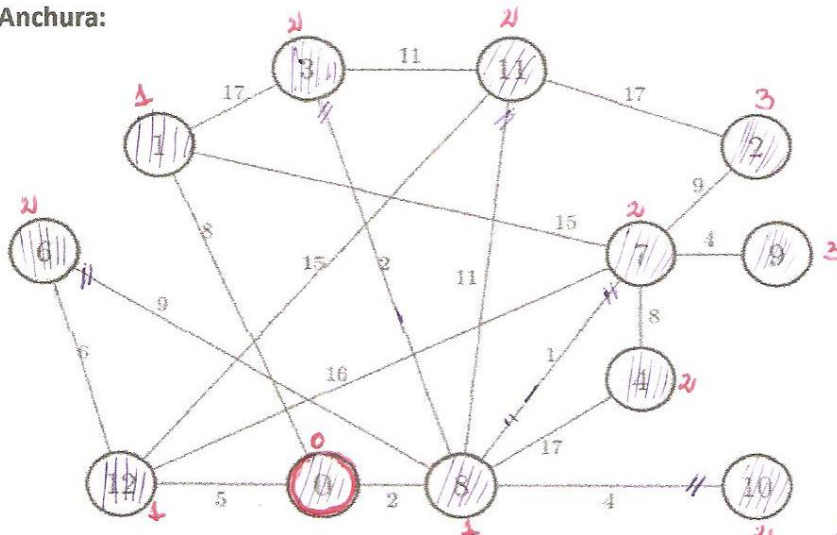


1	6
3	12
11	
9	42
7	10
8	
0	

Pila

Salida: 9-4-1-3-6-12-11-2-7-10-8-0

Anchura:



Cola:

0	8	12	1	7	3	10	6	11	4	9	2
1	1	1	2	2	2	2	2	2	2	3	3

Salida: 0-8-12-1-7-3-10-6-11-4-9-2