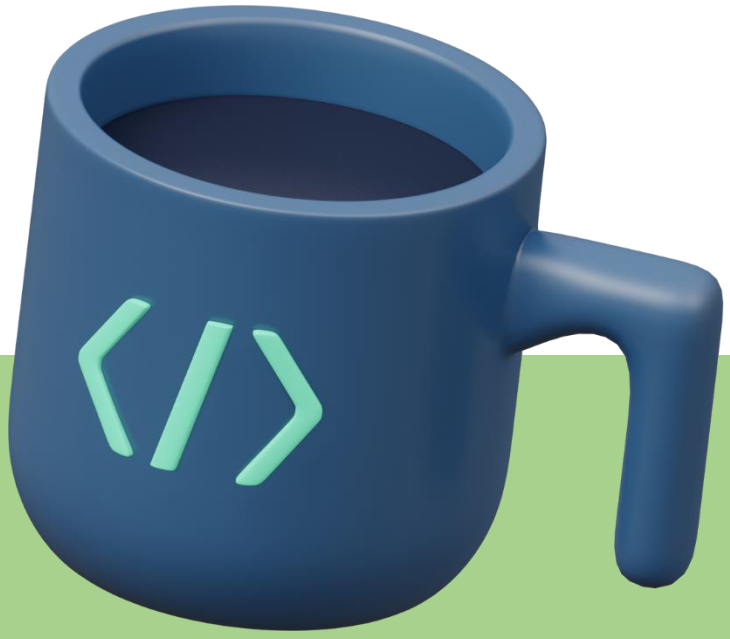
/TheVampi

02 Laboratorio Mandar el lab 01 a JAVA

Estructura de Datos

Martinez Rivera Luis Fernando

“Nunca pares de aprender”



Código fuente

Código completo disponible en el siguiente repositorio:

https://github.com/TheVampi/DataStructures-TecNM/tree/main/Java/Lab_02

```
package conjuntos;


public class Conjuntos {
    public static void main(String[] args) {
        int[] universo = {1,2,3,4,5,6,7,8,9,10,11,12,13,14};
        int[] A = {2,4,6,8,10,12,14};
        int[] B = {1,2,3,5,8,13};
        int[] C = {1,2,4,6,7,10,11,13};


        // #1 A union B
        System.out.print("A union B: ");
        for (int i = 0; i < A.length; i++) {
            System.out.print(A[i] + " ");
        }
        for (int i = 0; i < B.length; i++) {
            boolean found = false;
            for (int j = 0; j < A.length; j++) {
                if (B[i] == A[j]) {
                    found = true;
                    break;
                }
            }
            if (!found) {
```

```
            System.out.print(B[i] + " ");
        }
    }
    System.out.println();

    // #2 B interseccion C
    System.out.print("B interseccion C: ");
    for (int i = 0; i < B.length; i++) {
        for (int j = 0; j < C.length; j++) {
            if (B[i] == C[j]) {
                System.out.print(B[i] + " ");
                break;
            }
        }
    }
    System.out.println();

    // #3 Complemento de C
    System.out.print("Complemento de C: ");
    for (int i = 0; i < universo.length; i++) {
        boolean found = false;
        for (int j = 0; j < C.length; j++) {
            if (universo[i] == C[j]) {
                found = true;
                break;
            }
        }
        if (!found) {
            System.out.print(universo[i] + " ");
```

```
    }
}
System.out.println();

// #4 B union C
System.out.print("B union C: ");
for (int i = 0; i < B.length; i++) {
    System.out.print(B[i] + " ");
}
for (int i = 0; i < C.length; i++) {
    boolean found = false;
    for (int j = 0; j < B.length; j++) {
        if (C[i] == B[j]) {
            found = true;
            break;
        }
    }
    if (!found) {
        System.out.print(C[i] + " ");
    }
}
System.out.println();

// #5 A interseccion C
System.out.print("A interseccion C: ");
for (int i = 0; i < A.length; i++) {
    for (int j = 0; j < C.length; j++) {
        if (A[i] == C[j]) {
            System.out.print(A[i] + " ");
```

```
            break;
        }
    }
}
System.out.println();

// #6 Complemento de A
System.out.print("Complemento de A: ");
for (int i = 0; i < universo.length; i++) {
    boolean found = false;
    for (int j = 0; j < A.length; j++) {
        if (universo[i] == A[j]) {
            found = true;
            break;
        }
    }
    if (!found) {
        System.out.print(universo[i] + " ");
    }
}
System.out.println();

// #7 B menos A
System.out.print("B menos A: ");
for (int i = 0; i < B.length; i++) {
    boolean found = false;
    for (int j = 0; j < A.length; j++) {
        if (B[i] == A[j]) {
            found = true;
```

```
            break;
        }
    }
    if (!found) {
        System.out.print(B[i] + " ");
    }
}
System.out.println();

// #8 C menos el complemento de A
int[] complementoA= new int[universo.length-A.length];
int index=0;
for(int k=0;k<universo.length;k++){
    boolean found=false;
    for(int l=0;l<A.length;l++){
        if(universo[k]==A[l]){
            found=true;
            break;
        }
    }
    if(!found){
        complementoA[index]=universo[k];
        index++;
    }
}
System.out.print("C menos el complemento de A: ");
for(int i=0;i<C.length;i++){
    boolean found=false;
    for(int j=0;j<complementoA.length;j++){
```

```
            if(C[i]==complementoA[j]){
                found=true;
                break;
            }
        }
        if(!found){
            System.out.print(C[i]+" ");
        }
    }
    System.out.println();
  }
}
```

Capturas de ejecución