# Tarea 3 – Análisis de Algoritmos

## Vargas Aldaco Alejandro

El archivo se llama AUB.java y se compila en la terminal con javac AUB.java

```
3. AUB() {
        for(a:A)
                c=0
                for(b:B)
                        if(a==b) c++;
                enfor
                if(c>0) AUB.add(b)
        enfor
  }
```

Precondición todos los elementos de A, A y B son conjuntos de enteros
Postcondición AUB -> Todos los elementos de A y los elementos de B que no están en A

Sea c el invariante del for anidado. Entonces veamos que para el inicio de la primer iteración c = 0. Dado que si a !=b -> c=0. Si a == b entonces c = c + 1, lo cual es el invariante de la siguiente iteración, por lo tanto, el invariante se cumple en todos los ciclos del for anidado.

Sea AUB’ el invariante del segundo for.

Para el caso base. Antes de la primer iteración tenemos AUB’ = AU{/0} = A Dado que no se ha comparado ningún elemento de B con ningún elemento de A, por lo tanto se cumple el invariante.

H.I. Por definición AUB’ = {a,…,an} U {b1’,…,bj’} tal que bi’ es elemento de B’ y bi’ no es elemento de A para todo bi’;  1<= i <= j. Y AUB = {a,…,an} U {b1,…,bm}

Suponemos que se cumple el invariante en la (k)-ésima iteración 0<=k<m entonces el invariante es AUB’= {a,…,an} U {b1,…,j} j<=k  por la precondición

Pd se cumple para la (k+1)-ésima iteración entonces al inicio tenemos que el invariante es: AUB’= {a,…,an} U {b1,…,bj}.

Si el elemento k+1 de B está en A y al final de la iteración como ya se recorrieron todos los elementos de B tenemos AUB = AUB’ por lo tanto se cumple la postcondición.

Si el elemento k+1 de B no está en A, tenemos que al final de la iteración AUB'= {a,…,an} U {b1,…,bj} U {bk+1}. Esto es AUB' = {a,…,an} U {b1,…,bk+1} como k<m => k+1 <=m;
Si k+1 = m, entonces por HI tenemos AUB' = {a,…,an} U {b1,…,bm} = AUB Por lo tanto se cumple la postcondición.
Podemos ver que para el primer invariante la condición de pertenencia está dado desde el primer elemento de a hasta el último. Esto es de 0 hasta A.length-1 por lo tanto el for es finito. Así mismo el segundo invariante, la condición de pertenencia esta dada desde la posición 0 de B hasta B.length-1 por lo tanto el for es finito.

Dada la postcondición podemos ver que A es finito y B es finito entonces AUB es finito, como los dos for son finitos podemos comparar todos los elementos de A y con todos los elementos de B en una cantidad de iteraciones finita. Por lo tanto el algoritmo es finito.