

Test Suite Trabajo Práctico 2

En este pequeño documento/repositorio desarrollaremos una pequeña suite de pruebas para realizar test sobre la implementación del trabajo práctico.

Lo que haremos es un simple test basado en propiedades. Es decir, utilizaremos propiedades conocidas de conjuntos con las operaciones a desarrollar en el trabajo para manipular conjuntos y observar sus resultados.

En particular nos basaremos en lo que se llama *álgebra de conjuntos*.

A su vez haremos lo que se denomina *Test de Caja Negra*. Es decir, los casos de pruebas no se realizan en base al código o estructuras utilizadas.

En este documento introducimos la idea e invitamos a los estudiantes a generar nuevos casos de pruebas basados en las diferentes propiedades de conjuntos.

Definiremos diferentes conjuntos que utilizaremos a lo largo de los test

$A = \{0, 1, 2, 3\}$

$B = \{x : -3 \leq x \leq 5\}$

$C = \{x : 3 \leq x \leq 7\}$

Conmutatividad de la Unión e Intersección

Sabemos que tanto la unión como la intersección son operaciones conmutativas. Por lo tanto dados dos conjuntos P, Q , $P \cup Q \equiv Q \cup P$. Podemos verlo con:

$AB = A \mid B$

$BA = B \mid A$

`imprimir AB`

`imprimir BA`

Deberíamos ver que son conjuntos equivalentes. Lo mismo podemos hacer con la intersección de conjuntos $A \cap B = A \& B$ y $B \cap A = B \& A$.

Propiedades Distributivas

Sabemos que para conjuntos P, Q, R , $P \cup (Q \cap R) \equiv (P \cup Q) \cap (P \cup R)$. Primero definimos unos conjuntos auxiliares:

$CA = C \mid A$

$CB = C \mid B$

$CA \cap B = C \mid A \cap B$

$CA \cap CB = C \mid A \& B$

`imprimir CA \cap B`

`imprimir CA \cap CB`

Lo mismo podemos lograr distribuyendo la intersección sobre la unión: sean P, Q, R conjuntos, $P \cap (Q \cup R) \equiv (P \cap Q) \cup (P \cap R)$

Propiedades de resta

Sean P, Q conjuntos, $P \cap Q \equiv P \setminus (P \setminus Q)$. Podemos verificarlo con

```
Rest1 = A - B
Rest2 = A - Rest1
imprimir Rest2
imprimir AiB
```

Ambos conjuntos deberían ser equivalentes.

Propiedades de complemento.

El complemento de conjuntos posee propiedades sobre la unión y la intersección de conjuntos conocidas como las leyes De Morgan:

- $(P \cap Q)^C \equiv P^C \cup Q^C$, donde podemos extraer el siguiente caso de prueba:

```
CompA = ~ A
CompB = ~ B
CAuCB = CompA | CompB
CompAiB = ~ AiB
imprimir CompAiB
imprimir CAuCB
```

Deberíamos observar que ambos conjuntos son equivalentes.

- $(P \cup Q)^C \equiv P^C \cap Q^C$, donde podemos extraer el siguiente caso de prueba:

```
CompA = ~ A
CompB = ~ B
CAiCB = CompA & CompB
CompAB = ~ AB
imprimir CompAB
imprimir CAiCB
```

Deberíamos observar que ambos conjuntos son equivalentes.

Automatización Rápida

Todas las propiedades de equivalencia en principio deberían imprimir el mismo conjunto, con la salvedad que es posible que dependiendo de la representación interna conjuntos equivalentes pueden imprimirse de distinta forma.

Podemos utilizar las herramientas de la terminal de Linux (o similares en Windows) para automatizar rápidamente estos chequeos. Podemos escribir dos archivos simulando las interacciones del usuario: Un archivo llamado `ladoIzq.test`:

```
A = {0,1,2,3}
B = {x : -3 <= x <= 5}
C = {x : 3 <= x <= 7}
```

```

CA = C | A
CB = C | B
CAiB = C | AiB
CAiCB = CA & CB
imprimir CAiB

```

Y uno llamado `ladoDer.test`:

```

A = {0,1,2,3}
B = {x : -3 <= x <= 5}
C = {x : 3 <= x <= 7}
CA = C | A
CB = C | B
CAiB = C | AiB
CAiCB = CA & CB
imprimir CAiCB

```

Asumiendo que el interprete es un binario llamado `interprete`, podemos ejecutar los siguientes comandos:

```

>cat ladoIzq.test | ./interprete > ladoIzq.res
>cat ladoDer.test | ./interprete > ladoDer.res
>diff ladoIzq.res ladoDer.res

```

Los primeros dos comandos lo que hacen ejecutar las operaciones de cada lado de las equivalencias y escribiendo los resultados en dos archivos, que finalmente con `diff` compararemos. El comando `diff` nos indicará las diferencias entre los archivos resultando, y llegado al caso el estudiante podrá evaluar si son equivalentes o no.

Notar que es posible que conjuntos que son equivalentes pueden ser impresos de distinta forma. Dependerá de la estructura interna elegida y de cómo son implementadas las funciones de unión, intersección, etc.