Unidad 1: Técnicas de Conteo

# Principios básicos

## Principio de la suma (o)

SI una actividad se puede construir en *t* pasos sucesivos y cada paso *p* puede realizarse de *p maneras. El número de actividades posibles diferentes es* ***n1\*n2\*…\*nt***.

## Principio de la multiplicación (y)

Si una primera tarea puede realizarse de *m* formas y la segunda de *n* formas y no pueden realizarse en simultáneo, entonces, para hacer cualquiera de ellas hay ***m+n*** formas.

# Permutación

Una permutación de elementos diferentes *x1, x2, …, xn* es un ***ordenamiento*** de los *n* elementos.

## Teorema 1

Existen ***n!*** permutaciones de *n* elementos.

*Demostración:* Una permutación de *n* elementos se construye en *n* pasos sucesivos (principio de la mult.): se elige el primer elemento, el segundo, …, el último. El primero se puede seleccionar de *n* maneras, el segundo de maneras, el tercero de maneras, y así sucesivamente. Entonces:

## Permutación de *r* en *n* (*P(n, r)*)

Una permutación *r* elementos seleccionados de un conjunto de elementos diferentes *x1, x2, …, xn* es un ***ordenamiento*** de *r* elementos de {*x1, x2, …, xn*}.

*Demostración:* El primer elemento se puede elegir de maneras. Se elige el primer elemento de *n* maneras, el segundo de . Se continúa hasta llegar al elemento *r* que se puede seleccionar de maneras. Por el principio de la multiplicación, el número de permutaciones *r* de un conjunto de *n* objetos distintos es:

# Combinaciones

Sea un conjunto con *n* elementos diferentes, una combinación *r* de *X* es una ***selección no ordenada*** de *r* elementos de *X*.

## Teorema 1

El número de combinaciones *r* de un conjunto de *n* objetos distintos es:

## Teorema 2

Una sucesión *S* de *n* artículos tiene *n1* objetos iguales del *tipo 1*, *n2* objetos iguales del *tipo 2*, …, *nt* objetos iguales del *tipo t*, donde . Entonces, el número de ordenamientos de *S* es:

*Demostración:* Se asignan posiciones a cada uno de los *n* artículos para crear un ordenamiento de *S*. Se pueden asignar posiciones a los *n1* objetos del tipo 1 de maneras. Luego, se pueden asignar posiciones a los *n2* artículos del tipo 2 de maneras, y así sucesivamente. Por el principio de la multiplicación, el número de ordenamiento es:

…

## Combinaciones con repetición

Si *X* es un conjunto que contiene *t* elementos, el número de ***selecciones no ordenadas*** de *k* elementos de *X*, ***con repeticiones***, es:

*Demostración:* Sea . Considerando los *k + t - 1* espacios

\_ \_ \_ … \_ \_

y *k + t - 1* símbolos que consisten en *k* símbolos *x* y *t - 1*  símbolos *z*. Cada colocación de estos símbolos en los espacios determina una selección:

* El número *n1* de *x* hasta encontrar la primera *z* representa la selección *n1 a1*;
* el número *n2* de *x* entre la primera y la segunda *z* representa la selección *n2 a2* ;
* y así sucesivamente.

Como hay maneras de seleccionar las posiciones para las *z*, también hay selecciones. Esto es igual a , el número de maneras de seleccionar las posiciones para las *x*; entonces existen:

***selecciones*** ***no ordenadas*** de *k* elementos de *X*, con repeticiones.

## Propiedades

# Coeficientes binomiales e identidades combinatorias

## Teorema binomial

Si *a* y *b* son variables y *n* un *Z+*:

En notación sigma:

## Teorema multinomial

Sean *n* y *t* naturales, el coeficiente de en el desarrollo de:

es

Unidad 2: Lenguaje - Máquinas de estados finitos

# Lenguaje

El símbolo representa un conjunto de símbolos finitos *no* vacío llamado ***alfabeto***. Usándolo de base, podemos construir *cadenas* con base a los símbolos de .

## Definiciones

Def. 1: Definición 1: Si es un alfabeto y , definimos las potencias de .

, donde *xy* denota la yuxtaposición de x e y.

Def. 2: Para cualquier alfabeto definimos , donde denota la cadena vacía (no consta de ningún símbolo de ).

Aunque *. E*ntonces, es necesario aclarar que:

* *.*

Def. 3: Si es cualquier alfabeto:

y

Aunque los conjuntos y son *infinitos*, los elementos de estos conjuntos son cadenas *finitas* de símbolos.

Def. 4: Dos cadenas y son igualessolo cuándo cada una está formada por el mismo número de símbolos de y los símbolos correspondientes en las cadenas coinciden idénticamente. Es decir, si y .

Def. 5: La longitud de es . Para

Def. 6: La concatenación de y , denotada como , es la cadena La concatenación con será: .

Def. 7: Para cualquier serán las potenciasde x:

Def. 8: Si y , entonces:

* x es prefijo de w (prefijopropio si )
* y es sufijo de w (sufijopropio si )

Def. 9: Si y , entonces es una subcadena. Si y , es una subcadenapropia.

Def. 10: Para un alfabeto dado , cualquier subconjunto de es un lenguaje sobre . Esto incluye al subconjunto .

Ya que los lenguajes son conjuntos, podemos formar la unión, intersección y diferencia simétrica de dos lenguajes.

Def. 11: Para un alfabeto y los lenguajes , la concatenación de A y B, denotada con *AB*, es

Def. 12: Para cualquier lenguaje podemos construir otros lenguajes de la sgte. manera:

1. y para cualquier
2. , la *clausura positiva de A.*
3. , la *clausura de Kleene de A*.

Teorema 1

Para un alfabeto , sean . Entonces:

1. .
2. .
3. .
4. .
5. .

Teorema 1

Sea un Σ un alfabeto, con lenguajes . Si , entonces

Teorema 2

Para un alfabeto Σ y los lenguajes ,

# Máquinas de estados finitos

Una ***máquina de estados finitos*** es una 5-upla , donde es el conjunto de estados internos de ; es el alfabeto de entrada de ; es el alabeto de salida de ; es la *función siguiente estado*; y es la *función de salida.*

La salida en el instante es y le sigue la transición de la máquina, en el instante , al siguiente estado interno, dado por .

Es posible representar y por medio de la *tabla de estados* o *tabla de transición* y visualmente con un *diagrama de estados*.

El solapamiento es el caso en que algunos caracteres en la cadena de entrada son caracteres de más de una salida.

## Tipos

### Reconocedoras o detectoras

Detectan patrones o secuencias determinadas en respuesta a las entradas recibidas. No proveen acciones de salida, transicionan desde un estado inicial a un estado final de “Éxito”.

### De retraso

La salida de esta máquina no depende del estado actual inmediato, si no de los estados alcanzados con un ***retraso de k pasos*** respecto a la entrada, donde . La salida en el instante depende del estado alcanzado en el instante .

## Definiciones

Def. 1: Sea una máquina de estados finitos.

1. Para , decimos que  ***se puede alcanzar desde*** si o si existe una cadena de entrada tal que .
2. Un estado es ***transitorio*** si para implica ; es decir, no existe tal que .
3. Un estado es ***sumidero*** si .
4. Sea . Si , entonces con es una ***submaquina*** de .
5. Una máquina es ***fuertemente conexa*** si para cualesquiera estados , podemos alcanzar desde .

Def. 2: Para una máquina de estados finitos , sean dos estados distintos en . La cadena de entrada más corta es una ***secuencia de transferencia (o transicion)*** desde a si:

1. , y
2. con