

Mathematics for Computer Science

Jakxel

1 de agosto de 2025

Índice

1. Lógica Proposicional	2
1.1. Conectivos Lógicos	2
1.2. Tablas de verdad	2
2. Teoría de Conjuntos	2
2.1. Operaciones	2
3. Teoría de Números	2
3.1. Congruencias	2
4. Combinatoria	2
4.1. Permutaciones y Combinaciones	2
5. Probabilidad discreta 1	3
5.1. Eventos	3
6. Grafos	3
6.1. Definición	3
6.2. Árboles	3
7. Código en C++	3
7.1. Ejemplo: Factorial Recursivo	3
7.2. Código desde archivo externo	3

1. Lógica Proposicional

1.1. Conectivos Lógicos

Definición 1.1. Un *conectivo lógico* es un operador que une dos o más proposiciones para formar una proposición compuesta. Ejemplos comunes: \neg , \wedge , \vee , \rightarrow , \leftrightarrow .

Fórmula clave

$$p \rightarrow q \equiv \neg p \vee q$$

1.2. Tablas de verdad

Ejemplo 1.1. Verifica la validez de la siguiente proposición: $((p \rightarrow q) \wedge (q \rightarrow r)) \rightarrow (p \rightarrow r)$

2. Teoría de Conjuntos

2.1. Operaciones

Definición 2.1. La *intersección* de dos conjuntos A y B es el conjunto de todos los elementos que pertenecen a ambos: $A \cap B$.

Fórmula clave

$$\neg(A \cup B) = \neg A \cap \neg B \quad \text{y} \quad \neg(A \cap B) = \neg A \cup \neg B$$

3. Teoría de Números

3.1. Congruencias

Definición 3.1. $a \equiv b \pmod{n}$ significa que n divide a $a - b$.

Teorema 3.1 (Pequeño Teorema de Fermat). Si p es un número primo y a no es divisible por p , entonces:

$$a^{p-1} \equiv 1 \pmod{p}$$

4. Combinatoria

4.1. Permutaciones y Combinaciones

Fórmula clave

Permutaciones sin repetición: $P(n, r) = \frac{n!}{(n-r)!}$

Fórmula clave

Combinaciones: $C(n, r) = \binom{n}{r} = \frac{n!}{r!(n-r)!}$

5. Probabilidad discreta 1

5.1. Eventos

Definición 5.1. Un *evento* es un subconjunto del espacio muestral. Por ejemplo, al lanzar un dado, obtener un número par es un evento: $\{2, 4, 6\}$.

Ejemplo 5.1. ¿Cuál es la probabilidad de obtener al menos una cara al lanzar dos monedas?

6. Grafos

6.1. Definición

Definición 6.1. Un *grafo* $G = (V, E)$ consiste en un conjunto de vértices V y un conjunto de aristas $E \subseteq V \times V$.

6.2. Árboles

Teorema 6.1. Todo árbol con n vértices tiene exactamente $n - 1$ aristas.

7. Código en C++

7.1. Ejemplo: Factorial Recursivo

Definición 7.1. En muchas estructuras algorítmicas, es común definir funciones recursivas para resolver problemas. Aquí mostramos un ejemplo del cálculo del factorial usando C++.

```
1 // C++: Factorial recursivo
2 #include <iostream>
3 using namespace std;
4
5 int factorial(int n) {
6     if (n <= 1) return 1;
7     return n * factorial(n - 1);
8 }
9
10 int main() {
11     int n = 5;
12     cout << "Factorial de " << n << " es: " << factorial(n) << endl;
13     return 0;
14 }
```

Ejemplo 7.1. El código anterior imprimirá: *Factorial de 5 es: 120*

7.2. Código desde archivo externo

```
1 // factorial.cpp
2 #include <iostream>
3 using namespace std;
4
5 int factorial(int n) {
6     if (n <= 1) return 1;
7     return n * factorial(n - 1);
8 }
9
```

```
10 int main() {  
11     int n = 5;  
12     cout << "Factorial de " << n << " es: " << factorial(n) << endl;  
13     return 0;  
14 }
```

hola espero que esten este una prueba de live server con latex