**Bitset Built-in Functions:**

1. **Constructor**

|  |  |  |
| --- | --- | --- |
| **Name** | **Details** |  |
| **Bitset<value>bits;** | bitset<8> bits; // Creates a bitset with 8 bits, initialized to zero by default |  |
| **bits.set(pos)** | bits.set(2); // Sets the bit at position 2 to 1 |  |
| **bits.reset(pos)** | bits.reset(pos); |  |
| **bits.flip(pos)** | bits.flip(5); // Flips the bit at position 5 |  |
| **bits.test(pos)** | bits.test(pos); |  |
| **bits[0] = 1;** | bits[0] = 1; // sets the bit at position 0 to 1; |  |
| **bits.count()** | int cnt = bits.count(); // count the number of set bit |  |
| **bits.to\_string()** | string val = bits.to\_string(); |  |
| **bits.all()** | bool allset = bits.all(); // if all bit is set(1) when return true |  |
| **bits.any()** | bool anyset = bits.any(); // if any bit is set then return true |  |
| **bits.none()** | bool noneset = bits.none(); // all bit is zero then return true; |  |
| **bits.size()** | int sz = bits.size(); |  |
| **bits.to\_ulong()** | unsigned long value = bits.to\_ulong();// Converts the bitset to an unsigned long |  |
| **bits.to\_ullong()** | unsigned long long val = bits.to\_ullong(); // Converts the bitset to an unsigned long long |  |
| **cout<<bits;** | cout << "Bitset: " << bits <<endl; // Output: 00100100 |  |
| **\_\_builtin\_popcount(a^b)** | int a = 6; // 110;  int b = 5; //  101;  // দুটি সংখ্যার বিটওয়াইজ এর মধ্যে যে কতগুলি বিট পার্থক্য আছে সেটা পোপকাউন্ট এর মাধ্যমে ফেরত দেয়।  int diff = \_\_builtin\_popcount(a ^ b); // output:2 |  |