**Callsign Lookup Algorithm Description**

The algorithm determines a callsign's DXCC entity by processing it through a sequence of checks, ordered from most specific to most general. The first rule that provides a definitive answer stops the process.

1. **Initial Cleanup:** The raw callsign is first standardized. It's converted to uppercase, and common non-location suffixes like /P (portable), /M (mobile), or - are removed to isolate the core callsign for lookup.
2. **Exact Callsign Match (Highest Priority):** The algorithm first checks if the cleaned callsign is an **exact match** for a special-case callsign defined in the country file (e.g., =W1AW). These entries override all other rules and are used for stations with unique operating locations or statuses. If an exact match is found, that entity is returned immediately.
3. **Special Rule Checks:** Next, a few hard-coded special rules are checked:
   * **Maritime Mobile:** If the callsign ends in /MM, it is identified as a maritime station with no fixed country, and the process stops.
   * **Guantanamo Bay (KG4):** The algorithm specifically checks if the callsign fits the KG4xx (two-letter suffix) pattern. If it does, it's assigned to Guantanamo Bay. All other KG4 callsigns are correctly identified as standard United States stations later in the process.
4. **Portable Operation Logic (/):** If the callsign contains a slash, it's treated as a portable operation. The logic determines the location based on established conventions:
   * **US/Canada Home Call:** If the part *before* the slash has the structure of a US or Canadian callsign (e.g., W1AW/KP4), the part *after* the slash is treated as the location.
   * **Other Home Call:** If the part *before* the slash does not have a US or Canadian structure (e.g., EA8/W1AW), it is treated as the location.
5. **General Prefix Lookup (Longest Match):** If the callsign is not an exact match or a special case, the algorithm performs its primary function: a "longest prefix match." It starts with the full callsign and checks if it's a known prefix. If not, it shortens the callsign by one character from the right and checks again, repeating until it finds the longest valid prefix in its database (e.g., VO2AC -> VO2A -> VO2). The entity for that longest matching prefix is then returned.
6. **Fallback:** If, after all of these checks, no entity can be determined, the callsign is assigned to an "Unknown" entity