**Design Algorithm for FloridianToothRecords**

1. **Start Program**
   * Initialize constants for the maximum family members and teeth.
   * Declare arrays to hold family names and three-dimensional tooth records.
2. **Display Welcome Message**
3. **Get Family Size**
   * Prompt the user to enter the number of family members.
   * Validate that the number is within the allowed range (1 to 6).
   * Store the family size.
4. **Record Family Data**
   * For each family member:
     + Prompt the user for the family member’s name.
     + Record both the upper and lower teeth for each member by:
       - Prompting for teeth data (each tooth type: Incisor I, Bicuspid B, or Missing M).
       - Validate that each tooth type entered is valid and that the number of teeth does not exceed the allowed maximum (8 per row).
5. **Display Menu and Handle User Choice**
   * Loop until the user chooses to exit:
     + Display menu options (P)rint, (E)xtract, (R)oot, (eX)it.
     + Get and validate the user’s choice.
     + Based on the choice, call the corresponding method:
       - **(P)**: Print Family Teeth Records
       - **(E)**: Extract Tooth
       - **(R)**: Report Root Canal Indices
       - **(X)**: Exit Program
6. **Print Family Teeth Records**
   * For each family member:
     + Display their name.
     + Print upper and lower teeth rows.
7. **Extract Tooth**
   * Prompt the user for a family member’s name.
   * Validate the name; if invalid, re-prompt.
   * Prompt for the tooth layer (Upper or Lower).
   * Prompt for the tooth number to extract.
   * Validate that the tooth number is within range and not already missing.
   * Mark the tooth as missing.
8. **Report Root Canal Indices**
   * Count the total number of each tooth type (Incisors, Bicuspids, Missing) for all family members.
   * Calculate the root canal indices using a quadratic equation:
     + Use counts to solve the equation I×x2+B×x−M=0I \times x^2 + B \times x - M = 0I×x2+B×x−M=0 for x values.
   * Display the root canal indices.
9. **End Program**
   * Display an exit message.