Personalized Nutrition Recommendation Engine:

a. Create Binary Search Tree for dietary profiles

b. Implement nutritional requirement matching

c. Support meal planning and dietary restrictions

TIME COMPLEXITY REPORT:

 **Inserting Users into BST**:

* For **n** users, the average insertion time is **O (log n)** as

The **height** (number of levels) of a balanced BST is approximately **log₂(n)**. When you insert or search for a node, you **start at the root and go down** the tree — worst case, you travel down to a leaf. So the number of comparisons/steps is at most **log₂(n)**.

 **Inorder Traversal of BST**:

* This operation takes **O (n)** where **n** is the number of nodes/users

 **Loading Meals from CSV**:

* This operation takes **O (m)**, where **m** is the number of meals in the CSV file.

 **Filtering Meals for Each User**:

* For each user profile, filtering meals takes **O (m)** as it traverse through every meal to filter them

 **Generating Meal Plan for Each User**:

* For each user profile, generating a meal plan takes **O (m)** to traverse through all meals in worse case to generate meal plan

**Final Overall Time Complexity:**

* **O(n log n + m + n \* m)**, where:
  + **n** is the number of user
  + **m** is the number of meals in the CSV file.

In most practical scenarios, **O (n \* m)** would be the dominating factor, especially when we are having dataset with high number of meals