Implementation of the n-ary tree:-

Created a node of type structure with integer data, character array, and two node pointers(i.e., one for the child node and one for the sibling node). Creating another node of type structure pointer. Following are the functions used in the program:-

1. Check\_l1():- This function is used to check the category list present at level 1 in the tree (since level 1 contains the category list). If the user enters any category name which already exists in the tree then it will show a message as “Category already exists” else it will add the new category to the existing list.
2. depth():- This function is used to print the depth of any node in the tree. It takes the root of the type node structure and the word from the user as a function parameter.
3. childcount():- Function to count the child nodes for any given node present in the tree.
4. degree():- Function to count the degree of a node in the tree. The degree of a node of a tree is the number of subtrees having this node as a root.
5. print():- Function to display the words in decreasing order of their frequency. It takes the root of the tree as the function parameter. Within the function, it’s using pointers of type structure node to traverse through the tree and print the words onto the output screen in decreasing order of the word frequency.
6. main():- The program execution starts from main() function. Within the main() function we have the code to take user input for the category list to make. Also, we add the words into each category as required by the user. The code for displaying the leaves is also written inside main() only. To perform all the operations the user can choose from the list of options available to him/her

OUTPUT: 





