

323-33 Project 4: Huffman Coding

Student: Mohammed T Chowdhury

Project Due date: 03/21/2021

Algorithm Steps for implementing TreeNode class

Steps:

1. **TreeNode()** constructor initializes all the member variables
2. **PrintNode()** method checks which variables are not null. If it is Null print NULL otherwise print the data in the format: (T's chStr, T's frequency, T's code, T's next chStr, T's left's chStr, T's right's chStr);

```
public TreeNode(String chStr, int frequency, String code,TreeNode left,TreeNode right, TreeNode next ){
    this.chStr = chStr;
    this.frequency = frequency;
    this.code = code;
    this.left = left;
    this.right = right;
    this.next = next;
}

public TreeNode(){
    // format: (T's chStr, T's frequency, T's code, T's next chStr, T's left's chStr, T's right's chStr);
    //one print per text
    public void printNode(TreeNode t, FileWriter debugFile){
        try{
            if(this.left!= null&& this.right!=null && this.next != null) { // t.next.chStr != null && t.left.chStr != null && t.right
                debugFile.write( str: "("+this.chStr + "," + this.frequency + "," + this.code +
                    "," + t.next.chStr + "," + t.left.chStr + "," + t.right.chStr +")\n");
            }else if(this.left!= null&& this.right!=null && this.next == null){
                debugFile.write( str: "("+this.chStr + "," + this.frequency + "," + this.code +
                    "," + "null" + "," + t.left.chStr + "," + t.right.chStr +")\n");
            }else if(this.left!= null&& this.right==null){
                debugFile.write( str: "("+this.chStr + "," + this.frequency + "," + this.code +
                    "," + t.next.chStr + "," + t.left.chStr + "," + "+" +")\n");
            }else if(this.left == null && this.right!=null){
                debugFile.write( str: "("+this.chStr + "," + this.frequency + "," + this.code +
                    "," + t.next.chStr + "," + "NULL" + "," + t.right.chStr +")\n");
            }else if(t.next != null){
                debugFile.write( str: "("+this.chStr + "," + this.frequency + "," + this.code +
                    "," + t.next.chStr + "," + "null" + "," + "null" +")\n");
            }else{
                debugFile.write( str: "("+this.chStr + "," + this.frequency + "," + this.code +
                    "," + "NULL,NULL,NULL" + ") \n");
            }
        }
    }
}
```

Algorithm Steps for implementing LinkedList class

Steps:

1. **LinkedList()** constructor sets up the member variables.
2. **InsertNewNode()** method checks if the list is empty. If it is , then make the inserted node listHeads next node
 - a. Or else, iterate through the list, to see if a node in the list is bigger than the node being inserted.

- b. If it is, then insert the new node in front of the bigger node/
 - c. Or else , put it at the end of the list.
3. The printList() method iterates through the linked list and calls printNode() on each node.

```
public LinkedList(){

    TreeNode dummy = new TreeNode( chStr: "DUMMY", frequency: 0, code: "", left: null, right: null, next: null);
    listHead = dummy;
    listTail = dummy;
    listTail.next = null;
    listHead.next = null;
    this.size = 0;
}

// Insert a newNode into the linked list, in ascending order. Re-use your
// old code from your previous project.
public void insertNewNode(TreeNode node){
    TreeNode curr = listHead;
    TreeNode temp = listHead;
    if(isEmpty()) {
        listHead.next = node;
        listTail = node;
        this.size++;
    }else{
        while(curr.next != null){
            if(node.getFrequency() < curr.next.getFrequency()){
                TreeNode tmp2 = curr.next;
                curr.next = node;
                node.next = tmp2;
                this.size++;
                break;
            }
            curr = curr.next;
        }
        curr = temp;
    }
}
```

```

// Call printNode for every node on the list from listHead to the end of the list
public void printList(LinkedList list,FileWriter debugFile){
    TreeNode curr = list.getListHead();
    TreeNode temp = list.getListHead();
    try {
        debugFile.write( str: "***PRINTING LINKED LIST*** \n");
        while (curr != null) {
            curr.printNode(curr, debugFile);
            curr = curr.next;
        }
        curr = temp;
    }catch (IOException e){
        System.out.println("error in printList");
    }
}

public boolean isEmpty(){
    if(listHead.next == null){
        return true;
    }
    return false;
}

```

Algorithm Steps for implementing BinaryTree class

Steps:

1. BinaryTree() constructor initializes the trees root node to the given node.
2. InOrderTraversal() iterates through the tree and checks left nodes, prints the current node, then checks right nodes.
3. preOrderTraversal() iterates through the tree and prints the current node, checks left nodes, then checks right nodes.
4. postOrderTraversal() iterates through the tree and checks left nodes, then checks right nodes, then prints the current node.
5. isLeaf() method checks if a nodes left and right nodes are null. If it is, then it is a leaf

```

public void preOrderTraversal(TreeNode root, FileWriter debugFile){
    if(root == null){
        return;
    }
    root.printNode(root,debugFile);
    preOrderTraversal(root.left,debugFile);
    preOrderTraversal(root.right,debugFile);
}

//on your own
public void inOrderTraversal(TreeNode root, FileWriter debugFile){
    if(root == null){
        return;
    }
    preOrderTraversal(root.left,debugFile);
    root.printNode(root,debugFile);
    preOrderTraversal(root.right,debugFile);
}

// on your own
public void postOrderTraversal(TreeNode root, FileWriter debugFile){
    if(root == null){
        return;
    }
    preOrderTraversal(root.left,debugFile);
    preOrderTraversal(root.right,debugFile);
    root.printNode(root,debugFile);
}

// returns true if both node's left and right are null, return false otherwise.
public boolean isLeaf(TreeNode node){
    if(node.left == null && node.right == null){
        return true;
    }
    return false;
}

```

Algorithm Steps for implementing HuffmanCoding class

Steps:

1. HuffmanCoding() constructor initializes each element of charCountAry[] to 0;
2. computeCharCounts() method iterates through the original Input file and counts how many times each character repeats.
3. printCountAry() method iterates through the charCountAry[] and print out all the characters that are in the input text
4. constructHuffmanLList() iterates through the charCountAry[] and :
 - a. Extracts the char value of each index
 - b. Keeps track of how many times each char occurred
 - c. Adds the extracted information in a new node and adds it to the linkedList.
5. constructHuffmanBinTree() method iterates through the linked list. As it iterates :

- a. The current nodes character sequence and frequency is added and combined.
 - b. The first node is set as the left node.
 - c. The second node is set as the right node.
 - d. The head of the updated node is set as the trees root node.
6. constructCharCode() method iterates through the binary tree and identifies each character's code.
 - a. This is done by appending 0 at every left traversal and 1 at every right traversal.
7. The encode() method is responsible for compressing the input text.
 - a. This is done by iterating through the input text and encoding it into binary
8. The Decode() checks if the encode() method functions properly.
 - a. This is done by traversing the binary tree. If the encoding encoded a 1, we traverse right. If it's a 0 we go right and extract the character value of each code.
9. The userInterface() asks the user if they want to encode the file. If they do, the encoding process begins , if they choose not to, then the program terminates.

```
public class HuffmanCoding {
    public int[] charCountAry = new int[256]; // a 1-D array to store the character counts.
    public String[] charCode = new String[256]; // a 1-D array to store the Huffman code table,

    public HuffmanCoding(){
        for(int i = 0; i < charCountAry.length; i++){
            charCountAry[i] = 0;
        }
    }

    // Read a character from input file, use (int) to get index, ascii code of the character;
    // charCountAry[index]++. You should know how to do this method.
    public void computeCharCounts(File inFile){
        try{
            Scanner scan = new Scanner(inFile);
            while (scan.hasNextLine()){
                String textLine = scan.nextLine();
                for(int i = 0; i < textLine.length(); i++){
                    int index = (int)textLine.charAt(i);
                    if(index >= 0 && index < 256){
                        charCountAry[index]++;
                    } //end of if
                } //end of for loop
            } //end of while
        } catch (FileNotFoundException e){
            System.out.println("exception in computeCharCounts");
        }
    }
}
```

```

public class HuffmanCoding {
    public int[] charCountAry = new int[256]; // a 1-D array to store the character counts.
    public String[] charCode = new String[256]; // a 1-D array to store the Huffman code table,

    public HuffmanCoding(){
        for(int i = 0; i < charCountAry.length; i++){
            charCountAry[i] = 0;
        }
    }

    // Read a character from input file, use (int) to get index, ascii code of the character;
    // charCountAry[index]++. You should know how to do this method.
    public void computeCharCounts(File inFile){
        try{
            Scanner scan = new Scanner(inFile);
            while (scan.hasNextLine()){
                String textLine = scan.nextLine();
                for(int i = 0; i < textLine.length(); i++){
                    int index = (int)textLine.charAt(i);
                    if(index >= 0 && index < 256){
                        charCountAry[index]++;
                    } //end of if
                } //end of for loop
            } //end of while
        } catch (FileNotFoundException e){
            System.out.println("exception in computeCharCounts");
        }
    }
}

```

//algorithm given

```

public BinaryTree constructHuffmanBinTree(LinkedList list, FileWriter debugFile){
    BinaryTree tree = new BinaryTree();
    try {
        debugFile.write( str: "\n \n *** CONSTRUCTING HUFFMAN BINARY TREE \n");
        //TreeNode curr = list.getListHead();
        TreeNode newNode ;
        TreeNode tmp = new TreeNode();
        while(list.getListHead().next.next != null){
            newNode = new TreeNode();
            newNode.chStr= list.getListHead().next.getChStr() + list.getListHead().next.next.chStr;
            newNode.frequency = list.getListHead().next.getFrequency() + list.getListHead().next.next.frequency;
            newNode.left = list.getListHead().next;
            newNode.right = list.getListHead().next.next;
            newNode.next = null;
            list.insertNewNode(newNode);
            tmp = newNode;
            list.getListHead().next = list.getListHead().next.next.next;
            list.printList(list, debugFile);
        }

        tree.root = tmp;
    } catch (IOException e){
        System.out.println("error in constructHuffmanBinTree");
    } catch (NullPointerException e){
        System.out.println("Null pointer exception in constructHuffmanBinTree");
    }

    return tree;
}

```



```

public void constructCharCode(TreeNode t, String inputCode){
    if(t.left == null && t.right == null){
        t.code = inputCode;
        int index = (int)t.chStr.charAt(0);
        charCode[index] = inputCode;
    }else{
        constructCharCode(t.left, inputCode+inputCode+"0");
        constructCharCode(t.right, inputCode+inputCode+"1");
    }
}

//algorithm given
public void encode(File originalFile, FileWriter compressedFile,FileWriter debugFile){
    try{
        Scanner scan = new Scanner(originalFile);
        String code="";
        while (scan.hasNextLine()){
            String textLine = scan.nextLine();
            for(int i =0; i < textLine.length();i++){
                char charIn = textLine.charAt(i);
                int index = (int) charIn;
                if(index >= 0 && index < 256) {
                    code = charCode[index];
                }
                if(code!=null){
                    debugFile.write( str: "char:"+ index + " " + code+ "\n");
                    compressedFile.write(code);
                }
            }
        }
    }catch (FileNotFoundException e){
        System.out.println("file not found to encode");
    }catch (IOException e){
        System.out.println("IOException in encode");
    }
}

```

```

public void decode(File compressedFile, FileWriter deCompressedFile,File orig ,TreeNode root){
    TreeNode spot = root;
    Scanner scan;
    try{
        scan = new Scanner(compressedFile);
        String textLine = scan.nextLine();
        for(int i =0; i < textLine.length();i++) {
            if(spot.left == null && spot.right == null) {
                deCompressedFile.write(spot.chStr);
                spot = root;
            }
            char oneBit =textLine.charAt(i);
            if(oneBit == '0'){
                spot = spot.left;
            }else if(oneBit == '1' ){
                spot = spot.right;
            }else{
                System.out.println("Error! The compress file contains invalid character!");
                return;
            }
        }
        if(spot.left != null && spot.right != null){
            System.out.println("Error: The compress file is corrupted!");
        }
    }catch (IOException e){
        System.out.println("IOException in decode");
    }
}

```

```

public void userInterface(FileWriter debugFile, TreeNode root) {
    String nameOrg,nameCompress,nameDeCompress;
    char yesNo;
    Scanner in = new Scanner(System.in);
    System.out.println("Compress File?");
    yesNo = in.nextLine().charAt(0);
    if(yesNo == 'N' || yesNo == 'n'){
        return;
    }else{
        System.out.println("Enter the name (without .txt ) of the file to be compressed");
        nameOrg = in.nextLine();
    }
    nameCompress = nameOrg + "_Compressed.txt";
    nameDeCompress = nameOrg + "_DeCompress.txt";
    nameOrg = nameOrg+".txt";

    try {
        File orgFile = new File(nameOrg);
        FileWriter compFile = new FileWriter(nameCompress);
        FileWriter deCompFile = new FileWriter(nameDeCompress);
        encode(orgFile,compFile,debugFile);
        compFile.close();
        File compressedFile = new File(nameCompress);
        decode(compressedFile,deCompFile,orgFile,root);

        deCompFile.close();
    }catch (IOException e){
        System.out.println("Exception in User Interface");
    }
}

```

Output: data.txt

Original:

The Old Man and the Sea tells the story of a battle between an aging, experienced fisherman, Santiago, and a large marlin. The story opens with Santiago having gone 84 days without catching a fish, and now being seen as *salao*, the worst form of unluckiness. He is so unlucky that his young apprentice, Manolin, has been forbidden by his parents to sail with him and has been told instead to fish with successful fishermen. The boy visits Santiago's shack each night, hauling his fishing gear, preparing food, talking about American baseball and his favorite player, Joe DiMaggio. Santiago tells Manolin that on the next day, he will venture far out into the Gulf Stream, north of Cuba in the Straits of Florida to fish, confident that his unlucky streak is near its end. On the eighty-fifth day of his unlucky streak, Santiago takes his skiff into the Gulf Stream, sets his lines and by noon, has his bait taken by a big fish that he is sure is a marlin. Unable to haul in the great marlin, Santiago is instead pulled by the marlin, and two days and nights pass with Santiago holding onto the line. Though wounded by the struggle and in pain, Santiago expresses a compassionate appreciation for his adversary, often referring to him as a brother. He also determines that, because of the fish's great dignity, no one shall deserve to eat the marlin. On the third day, the fish begins to circle the skiff. Santiago, worn out and almost delirious, uses all his remaining strength to pull the fish onto its side and stab the marlin with a harpoon. Santiago straps the marlin to the side of his skiff and heads home, thinking about the high price the fish will bring him at the market and how many people he will feed. On his way in to shore, sharks are attracted to the marlin's blood. Santiago kills a great mako shark with his harpoon, but he loses the weapon. He makes a new harpoon by strapping his knife to the end of an oar to help ward off the next line of sharks; five sharks are slain and many others are driven away. But the sharks keep coming, and by nightfall the sharks have almost devoured the marlin's entire carcass, leaving a skeleton consisting mostly of its backbone, its tail, and its head. Santiago knows that he is destroyed and tells the sharks of how they have killed his dreams. Upon reaching the shore before dawn on the next day, Santiago struggles to his shack, carrying the heavy mast on his shoulder, leaving the fish head and the bones on the shore. Once home, he slumps onto his bed and falls into a deep sleep. A group of fishermen gather the next day around the boat where the fish's skeleton is still

attached. One of the fishermen measures it to be 18 feet (5.5 m) from nose to tail. Pedrico is given the head of the fish, and the other fishermen tell Manolin to tell the old man how sorry they are. Tourists at the nearby cafe mistakenly take it for a shark. Manolin, worried about the old man, cries upon finding him safe asleep and at his injured hands, and brings him newspapers and coffee. When the old man wakes, they promise to fish together once again. Upon his return to sleep, Santiago dreams of his youth of lions on an African beach.

Data.txt Compressed:

```
110101111000001011010010001010011111111000011101001011101100100010100000010101011011110100111011
100101100100110011010000000101000000011010110000111101011110011111111110100100011010000000101000
100100101010110010011011010011111000111110100011100001100010100110000010001110110010110100000001
0010000011110001111001110011011101011100110111111001111101010011110111100111011100100000111011010
110001111101101110111111111000011100001000000011101101010001100010001110011100101100110010011101
10101111011101111110100101101000000010111010110001110011011100011100010100010111000101011000011
1111001110111100001101011100001101101010011101100111110000111010010111010000110000010010011101111
0001111110011110101010010110101001000010101011101000000010110000010100011111100100001001010011101
1000001110010011100110011010001110000111100110101000111101001011010001100110011010100110110011001
110011001000111001101111110111000110111000011101010010111111010011110011001100001111111001110011
0111111101110001101111010000000101001101110110111100110010010011100110011010001110001010001110111
111001110100001101110111011111001110101011100001001110000001101001100001110110001001010111000000
11110100000001011000111101011101111100001100010000011111000101111100001000110100000001010101101
011010111010101000101110111111001101010101100011110001101000000010110111100111010001110011110000
010001111011110011101110000110001100010010100110001111111100100111011001001101101010000111010111
001110001010011100011101101110000101100101100110101100101011100001011111100010100110100101010001
000011100110011111101100111011001011110001111100101010111100110011010000000101010000110000100111
0000111000010101110010111110101100100000111011000011010000000011111101101000000010101100111101000
0111010110011011110011000010001011011010100111011001011111001111100101011000100011101011010100100
00111101110000100000111111110101111010100100011001010010000011111011100001110110001110000110011
100101000011110101101110111010000000101110110001010111000001101100011110110100000011111011101101
0100100010001111010001101110001001000000111111011101111101100111101111001011011110010101010000
00011101111001000000111100110010101010000011110110100111111100010010100001111011000111000000011
111100110001110011010010100001000001111111010100001000011110111101110101111011100010100111011111
0101110000100001111011000100101000100100111010000111101011010011101110000011111000010000100100
101001101110101110001011111110011101001110011011111011000001000110100000001011101101000001010011
1111110001010000111101111000100010111110001001010111001110000100110011110011100011100010110011
1011101101100011100001000101000111111101000000010101101010101100011100110111110000011011111110101
1010000100101111100100111000010010010110110010000010111011010011101110010101011110111011000110000
001010111101010111100101011111011001100100111001010010011101000111110001110111010001000100111010
1100110101101101110100000111001010110010110010011101000011101110110010011110101001101111001010010
01110100100110100100111010010011101111000111000110101010110011111000101010101010001100101100001
01001111001001110100010100001000011011111110111110001110110101110110010100001111101111111100001
1001000110111011111001101110110001100000111101100010000110111000111011000111000111011110110001010
01100100010101111000001101010111011100000111010110110010001111000111101001100111001100110010101
10101011010100001000001101110101111111100111011000101101001111100100001010001001111001111001110011
1010111101101000000010101110001100111111100001100011111000011101100011100
```

Data.txt DeCompressed:

The Old Man and the Sea tells the story of a battle between an aging, experienced fisherman, Santiago, and a large marlin. The story opens with Santiago having gone 84 days without catching a fish, and now being seen as *salao*, the worst form of unluckiness. He is so unlucky that his young apprentice, Manolin, has been forbidden by his parents to sail with him and has been told instead to fish with successful fishermen. The boy visits Santiago's shack each night, hauling his fishing gear, preparing food, talking about American baseball and his favorite player, Joe DiMaggio. Santiago tells Manolin that on the next day, he will venture far out into the Gulf Stream, north of Cuba in the Straits of Florida to fish, confident that his unlucky streak is near its end. On the eighty-fifth day of his unlucky streak, Santiago takes his skiff into the Gulf Stream, sets his lines and by noon, has his bait taken by a big fish that he is sure is a marlin. Unable to haul in the great marlin, Santiago is instead pulled by the marlin, and two days and nights pass with Santiago holding onto the line. Though wounded by the struggle and in pain, Santiago expresses a compassionate appreciation for his adversary, often referring to him as a brother. He also determines that, because of the fish's great dignity, no one shall deserve to eat the marlin. On the third day, the fish begins to circle the skiff. Santiago, worn out and almost delirious, uses all his remaining strength to pull the fish onto its side and stab the marlin with a harpoon. Santiago straps the marlin to the side of his skiff and heads home, thinking about the high price the fish will bring him at the market and how many people he will feed. On his way in to shore, sharks are attracted to the marlin's blood. Santiago kills a great mako shark with his harpoon, but he loses the weapon. He makes a new harpoon by strapping his knife to the end of an oar to help ward off the next line of sharks; five sharks are slain and many others are driven away. But the sharks keep coming, and by nightfall the sharks have almost devoured the marlin's entire carcass, leaving a skeleton consisting mostly of its backbone, its tail, and its head. Santiago knows that he is destroyed and tells the sharks of how they have killed his dreams. Upon reaching the shore before dawn on the next day, Santiago struggles to his shack, carrying the heavy mast on his shoulder, leaving the fish head and the bones on the shore. Once home, he slumps onto his bed and falls into a deep sleep. A group of fishermen gather the next day around the boat where the fish's skeleton is still attached. One of the fishermen measures it to be 18 feet (5.5 m) from nose to tail. Pedricco is given the head of the fish, and the other fishermen tell Manolin to tell the old man how sorry they are. Tourists at the nearby cafe mistakenly take it for a shark. Manolin, worried about the old man, cries upon finding him safe asleep and at his injured hands, and brings him newspapers and coffee. When the old man wakes, they promise to fish together once again. Upon his return to sleep, Santiago dreams of his youth of lions on an African beach.

Data.txt Debug:

PRINTING CHARACTER COUNT ARRAY ** >>>**

char32 578

char39 5

char40 1

char41 1

char44 43

char45 1

char46 27

char49 1

char52 1

char53 2

char56 2

char59 1

char65 3

char66 1

char67 1

char68 1
char70 1
char71 2
char72 3
char74 1
char77 6
char79 6
char80 1
char83 18
char84 5
char85 3
char87 1
char97 241
char98 40
char99 38
char100 84
char101 270
char102 71
char103 65
char104 184
char105 196
char106 1
char107 37
char108 101
char109 49
char110 199
char111 171
char112 40
char114 126
char115 191
char116 220
char117 46
char118 15
char119 30
char120 6
char121 40

PRINTING LINKED LIST

```

(DUMMY,0,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,' ,null,null)
(' ,5,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,( ,null,null)
(( ,1,, ,null,null)
(' ,5,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,( ,null,null)
(( ,1,, ),null,null)
( ),1,, ,null,null)
(' ,5,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,( ,null,null)
(( ,1,, ),null,null)
( ),1,, ,null,null)
(' ,5,,,,null,null)
( ,43,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,( ,null,null)
(( ,1,, ),null,null)
( ),1,, -,null,null)
(- ,1,, ,null,null)
(' ,5,,,,null,null)
( ,43,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,( ,null,null)
(( ,1,, ),null,null)
( ),1,, -,null,null)
(- ,1,, ,null,null)

```

```

('5,,,null,null)
(.,27,,,,null,null)
(.,43,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,(,null,null)
((1,,),null,null)
(),1,-,null,null)
(-,1,,1,null,null)
(1,1,',null,null)
('5,,,null,null)
(.,27,,,,null,null)
(.,43,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,(,null,null)
((1,,),null,null)
(),1,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,',null,null)
('5,,,null,null)
(.,27,,,,null,null)
(.,43,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,(,null,null)
((1,,),null,null)
(),1,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,5,null,null)
(5,2,',null,null)
('5,,,null,null)
(.,27,,,,null,null)
(.,43,, ,null,null)
( ,578,,NULL,NULL,NULL)

```

*****PRINTING LINKED LIST*****

(DUMMY,0,,(null,null))

((1,,),null,null)

(,1,-,null,null)

(-,1,,1,null,null)

(1,1,,4,null,null)

(4,1,,5,null,null)

(5,2,,8,null,null)

(8,2,,',null,null)

(',5,,,null,null)

(.,27,,,null,null)

(,,43,, ,null,null)

(,578,,NULL,NULL,NULL)

*****PRINTING LINKED LIST*****

(DUMMY,0,,(null,null))

((1,,),null,null)

(,1,-,null,null)

(-,1,,1,null,null)

(1,1,,4,null,null)

(4,1,,,null,null)

(:,1,,5,null,null)

(5,2,,8,null,null)

(8,2,,',null,null)

(',5,,,null,null)

(.,27,,,null,null)

(,,43,, ,null,null)

(,578,,NULL,NULL,NULL)

*****PRINTING LINKED LIST*****

(DUMMY,0,,(null,null))

((1,,),null,null)

(,1,-,null,null)

(-,1,,1,null,null)

(1,1,,4,null,null)

(4,1,,,null,null)

(:,1,,5,null,null)

(5,2,,8,null,null)

(8,2,,A,null,null)


```
(A,3,,',null,null)
(',5,,,null,null)
(.,27,,,,null,null)
(.,43,,',null,null)
(.,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
```

```
(DUMMY,0,,(,null,null)
((,1,,),null,null)
(,1,,-',null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,:',null,null)
(,1,,B,null,null)
(B,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,A,null,null)
(A,3,,',null,null)
(',5,,,null,null)
(.,27,,,,null,null)
(.,43,,',null,null)
(.,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
```

```
(DUMMY,0,,(,null,null)
((,1,,),null,null)
(,1,,-',null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,:',null,null)
(,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,A,null,null)
(A,3,,',null,null)
(',5,,,null,null)
(.,27,,,,null,null)
(.,43,,',null,null)
```

(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,(,null,null)

((,1,,),null,null)

(,1,,-,null,null)

(-,1,,1,null,null)

(1,1,,4,null,null)

(4,1,,,;null,null)

(;,1,,B,null,null)

(B,1,,C,null,null)

(C,1,,D,null,null)

(D,1,,5,null,null)

(5,2,,8,null,null)

(8,2,,A,null,null)

(A,3,,',null,null)

(',5,,,null,null)

(.,27,,,,null,null)

(,,43,, ,null,null)

(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,(,null,null)

((,1,,),null,null)

(,1,,-,null,null)

(-,1,,1,null,null)

(1,1,,4,null,null)

(4,1,,,;null,null)

(;,1,,B,null,null)

(B,1,,C,null,null)

(C,1,,D,null,null)

(D,1,,F,null,null)

(F,1,,5,null,null)

(5,2,,8,null,null)

(8,2,,A,null,null)

(A,3,,',null,null)

(',5,,,null,null)

(.,27,,,,null,null)

(,,43,, ,null,null)

(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,(,null,null)

((,1,,),null,null)

(,1,,-,null,null)

(-,1,,1,null,null)

(1,1,,4,null,null)

(4,1,,,;null,null)

(;,1,,B,null,null)

(B,1,,C,null,null)

(C,1,,D,null,null)

(D,1,,F,null,null)

(F,1,,5,null,null)

(5,2,,8,null,null)

(8,2,,G,null,null)

(G,2,,A,null,null)

(A,3,,',null,null)

(',5,,,null,null)

(.,27,,,null,null)

(,43,, ,null,null)

(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,(,null,null)

((,1,,),null,null)

(,1,,-,null,null)

(-,1,,1,null,null)

(1,1,,4,null,null)

(4,1,,,;null,null)

(;,1,,B,null,null)

(B,1,,C,null,null)

(C,1,,D,null,null)

(D,1,,F,null,null)

(F,1,,5,null,null)

(5,2,,8,null,null)

(8,2,,G,null,null)

(G,2,,A,null,null)

(A,3,,H,null,null)

```
(H,3,,',null,null)
('5,,,,null,null)
(.,27,,,,null,null)
(.,43,,',null,null)
(.,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
```

```
(DUMMY,0,,(,null,null)
```

```
((1,,),null,null)
```

```
(,1,-,null,null)
```

```
(-,1,1,null,null)
```

```
(1,1,4,null,null)
```

```
(4,1,,,null,null)
```

```
(;,1,,B,null,null)
```

```
(B,1,,C,null,null)
```

```
(C,1,,D,null,null)
```

```
(D,1,,F,null,null)
```

```
(F,1,,J,null,null)
```

```
(J,1,,5,null,null)
```

```
(5,2,,8,null,null)
```

```
(8,2,,G,null,null)
```

```
(G,2,,A,null,null)
```

```
(A,3,,H,null,null)
```

```
(H,3,,',null,null)
```

```
('5,,,,null,null)
```

```
(.,27,,,,null,null)
```

```
(.,43,,',null,null)
```

```
(.,578,,NULL,NULL,NULL)
```

```
***PRINTING LINKED LIST***
```

```
(DUMMY,0,,(,null,null)
```

```
((1,,),null,null)
```

```
(,1,-,null,null)
```

```
(-,1,1,null,null)
```

```
(1,1,4,null,null)
```

```
(4,1,,,null,null)
```

```
(;,1,,B,null,null)
```

```
(B,1,,C,null,null)
```

```
(C,1,,D,null,null)
```

```

(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,',null,null)
(',5,,M,null,null)
(M,6,,,null,null)
(.,27,,,null,null)
(.,43,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,(,null,null)
((,1,,),null,null)
(),1,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(.,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,',null,null)
(',5,,M,null,null)
(M,6,,O,null,null)
(O,6,,,null,null)
(.,27,,,null,null)
(.,43,, ,null,null)
( ,578,,NULL,NULL,NULL)

```

*****PRINTING LINKED LIST*****

(DUMMY,0,,(null,null))

((1,,),null,null)

(,1,-,null,null)

(-,1,,1,null,null)

(1,1,,4,null,null)

(4,1,,,null,null)

(;,1,,B,null,null)

(B,1,,C,null,null)

(C,1,,D,null,null)

(D,1,,F,null,null)

(F,1,,J,null,null)

(J,1,,P,null,null)

(P,1,,5,null,null)

(5,2,,8,null,null)

(8,2,,G,null,null)

(G,2,,A,null,null)

(A,3,,H,null,null)

(H,3,,',null,null)

(',5,,M,null,null)

(M,6,,O,null,null)

(O,6,,,null,null)

(.,27,,,,null,null)

(,,43,, ,null,null)

(,578,,NULL,NULL,NULL)

*****PRINTING LINKED LIST*****

(DUMMY,0,,(null,null))

((1,,),null,null)

(,1,-,null,null)

(-,1,,1,null,null)

(1,1,,4,null,null)

(4,1,,,null,null)

(;,1,,B,null,null)

(B,1,,C,null,null)

(C,1,,D,null,null)

(D,1,,F,null,null)

(F,1,,J,null,null)


```

(J,1,,P,null,null)
(P,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,',null,null)
(',5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,,null,null)
(.,27,,,null,null)
(.,43,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,(null,null)
((1,,),null,null)
(),1,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(.,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)

```

```

(S,18,, ,null,null)
(.,27,,,,null,null)
(.,43,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,(,null,null)
((,1,,),null,null)
(),1,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,, ,null,null)
(.,27,,,,null,null)
(.,43,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,(,null,null)
((,1,,),null,null)
(),1,-,null,null)
(-,1,,1,null,null)

```

(1,1,,4,null,null)
(4,1,,,null,null)
(,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,,null,null)
(.,27,,,null,null)
(,,43,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,(,null,null)
((,1,,),null,null)
(,1,,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)

(P,1,,W,null,null)
(W,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,,null,null)
(.,27,,,,null,null)
(.,43,,a,null,null)
(a,241,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,(,null,null)
((,1,,),null,null)
(,1,,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(;,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)

```
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,',null,null)
(.,27,,b,null,null)
(b,40,,,,null,null)
(.,43,,a,null,null)
(a,241,,',null,null)
(.,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,(,null,null)
((,1,,),null,null)
(),1,,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,,null,null)
(.,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,',null,null)
```

```
(.,27,,c,null,null)
(c,38,,b,null,null)
(b,40,,,,null,null)
(.,43,,a,null,null)
(a,241,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,(,null,null)
((,1,,),null,null)
(),1,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(.,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,,null,null)
(.,27,,c,null,null)
(c,38,,b,null,null)
(b,40,,,,null,null)
(.,43,,d,null,null)
(d,84,,a,null,null)
```



```
(a,241,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,(,null,null)
((,1,,),null,null)
(),1,,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,;,null,null)
(;,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,;,null,null)
(.,27,,c,null,null)
(c,38,,b,null,null)
(b,40,,,,null,null)
(.,43,,d,null,null)
(d,84,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
```

```

(DUMMY,0,,(,null,null)
((,1,,),null,null)
(),1,,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,,null,null)
(.,27,,c,null,null)
(c,38,,b,null,null)
(b,40,,,,null,null)
(.,43,,f,null,null)
(f,71,,d,null,null)
(d,84,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,(,null,null)
((,1,,),null,null)

```

```
(,1,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,,null,null)
(.,27,,c,null,null)
(c,38,,b,null,null)
(b,40,,,,null,null)
(,43,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,(,null,null)
((,1,,),null,null)
(),1,-,null,null)
```

(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,,null,null)
(.,27,,c,null,null)
(c,38,,b,null,null)
(b,40,,,,null,null)
(,,43,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,h,null,null)
(h,184,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,(,null,null)
((,1,,),null,null)
(,1,,-,null,null)

```

(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,,null,null)
(.,27,,c,null,null)
(c,38,,b,null,null)
(b,40,,,,null,null)
(.,43,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,h,null,null)
(h,184,,i,null,null)
(i,196,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,(,null,null)
((,1,,),null,null)

```

(,1,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(:,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,j,null,null)
(j,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,,null,null)
(.,27,,c,null,null)
(c,38,,b,null,null)
(b,40,,,,null,null)
(,,43,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,h,null,null)
(h,184,,i,null,null)
(i,196,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,(,null,null)
(,1,,),null,null)
(,1,,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(;,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,j,null,null)
(j,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,,null,null)
(.,27,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,,,null,null)
(,,43,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,h,null,null)
(h,184,,i,null,null)
(i,196,,a,null,null)
(a,241,,e,null,null)

(e,270,, ,null,null)

(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,(,null,null)

((,1,,),null,null)

(,1,-,null,null)

(-,1,,1,null,null)

(1,1,,4,null,null)

(4,1,,,null,null)

(;,1,,B,null,null)

(B,1,,C,null,null)

(C,1,,D,null,null)

(D,1,,F,null,null)

(F,1,,J,null,null)

(J,1,,P,null,null)

(P,1,,W,null,null)

(W,1,,j,null,null)

(j,1,,5,null,null)

(5,2,,8,null,null)

(8,2,,G,null,null)

(G,2,,A,null,null)

(A,3,,H,null,null)

(H,3,,U,null,null)

(U,3,,',null,null)

(',5,,T,null,null)

(T,5,,M,null,null)

(M,6,,O,null,null)

(O,6,,S,null,null)

(S,18,,,null,null)

(.,27,,k,null,null)

(k,37,,c,null,null)

(c,38,,b,null,null)

(b,40,,,,null,null)

(,,43,,g,null,null)

(g,65,,f,null,null)

(f,71,,d,null,null)

(d,84,,l,null,null)

```
(l,101,,h,null,null)
(h,184,,i,null,null)
(i,196,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,(,null,null)
((,1,,),null,null)
(),1,,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,j,null,null)
(j,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,,null,null)
(.,27,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,,,null,null)
```

(,,43,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,h,null,null)
(h,184,,i,null,null)
(i,196,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,(,null,null)
((,1,,),null,null)
(,1,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(:,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,j,null,null)
(j,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)

(S,18,,,null,null)
(.,27,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,,,null,null)
(.,43,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,h,null,null)
(h,184,,i,null,null)
(i,196,,n,null,null)
(n,199,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,(,null,null)
((,1,,),null,null)
(,1,,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(;,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,j,null,null)
(j,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)

(H,3,,U,null,null)
(U,3,',' ,null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,, ,null,null)
(.,27,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,,,null,null)
(, ,43,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,o,null,null)
(o,171,,h,null,null)
(h,184,,i,null,null)
(i,196,,n,null,null)
(n,199,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,(' ,null,null)
((,1,,),null,null)
(,1,, -,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,, ,null,null)
(;,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)

(P,1,,W,null,null)
(W,1,,j,null,null)
(j,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,,null,null)
(.,27,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,,,null,null)
(.,43,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,o,null,null)
(o,171,,h,null,null)
(h,184,,i,null,null)
(i,196,,n,null,null)
(n,199,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)
PRINTING LINKED LIST
(DUMMY,0,,(,null,null)
((,1,,),null,null)
(,1,,-,null,null)
(-,1,,1,null,null)

(1,1,,4,null,null)
(4,1,,,null,null)
(;,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,j,null,null)
(j,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,,null,null)
(.,27,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,,null,null)
(,43,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,i,null,null)
(i,196,,n,null,null)


```
(n,199,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,(,null,null)
((,1,,),null,null)
(),1,,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,j,null,null)
(j,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,,null,null)
(.,27,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,,null,null)
(.,43,,m,null,null)
```

(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,(null,null))
((1,,),null,null)
(,1,,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,;,null,null)
(;,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,j,null,null)
(j,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)

(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,,null,null)
(.,27,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,,,null,null)
(.,43,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,(,null,null)
((,1,,),null,null)
(,1,, -,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(;,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)

(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,j,null,null)
(j,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,S,null,null)
(S,18,,,null,null)
(.,27,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,,null,null)
(,,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

*****PRINTING LINKED LIST*****

(DUMMY,0,,(null,null)

((,1,,),null,null)

(,1,,-,null,null)

(-,1,,1,null,null)

(1,1,,4,null,null)

(4,1,,,null,null)

(;,1,,B,null,null)

(B,1,,C,null,null)

(C,1,,D,null,null)

(D,1,,F,null,null)

(F,1,,J,null,null)

(J,1,,P,null,null)

(P,1,,W,null,null)

(W,1,,j,null,null)

(j,1,,5,null,null)

(5,2,,8,null,null)

(8,2,,G,null,null)

(G,2,,A,null,null)

(A,3,,H,null,null)

(H,3,,U,null,null)

(U,3,,',null,null)

(',5,,T,null,null)

(T,5,,M,null,null)

(M,6,,O,null,null)

(O,6,,v,null,null)

(v,15,,S,null,null)

(S,18,,,null,null)

(.,27,,k,null,null)

(k,37,,c,null,null)

(c,38,,b,null,null)

(b,40,,p,null,null)

(p,40,,,null,null)

(,,43,,u,null,null)

(u,46,,m,null,null)

(m,49,,g,null,null)

(g,65,,f,null,null)

(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,(,null,null)
((,1,,),null,null)
(,1,,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,;,null,null)
(;,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,j,null,null)
(j,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)

(M,6,,O,null,null)
(O,6,,v,null,null)
(v,15,,S,null,null)
(S,18,,,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,,,null,null)
(,,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,(,null,null)
((,1,,),null,null)
(,1,,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(:,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)

(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,j,null,null)
(j,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,x,null,null)
(x,6,,v,null,null)
(v,15,,S,null,null)
(S,18,,,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,,null,null)
(,,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)


```
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,(,null,null)
((,1,,),null,null)
(),1,,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,;,null,null)
(;,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,j,null,null)
(j,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,x,null,null)
(x,6,,v,null,null)
(v,15,,S,null,null)
(S,18,,;,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
```

(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,,null,null)
(,,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

***** CONSTRUCTING HUFFMAN BINARY TREE**

*****PRINTING LINKED LIST*****

(DUMMY,0,-,null,null)
(-,1,,1,null,null)
(1,1,,4,null,null)
(4,1,,,null,null)
(;,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,j,null,null)

(j,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,(),null,null)
((),2,null,A,(,))
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,x,null,null)
(x,6,,v,null,null)
(v,15,,S,null,null)
(S,18,,,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,null,null)
(.,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)

(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)
PRINTING LINKED LIST
(DUMMY,0,,4,null,null)
(4,1,, ,null,null)
(:,1,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,j,null,null)
(j,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,(),null,null)
((),2,null,-1,(,))
(-1,2,null,A,-,1)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,x,null,null)
(x,6,,v,null,null)
(v,15,,S,null,null)
(S,18,, ,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,, ,null,null)
(.,43,,u,null,null)

(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,B,null,null)
(B,1,,C,null,null)
(C,1,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,j,null,null)
(j,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,(),null,null)
((),2,null,-1,(,))
(-1,2,null,4,,-,1)
(4,2,null,A,4,,:)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)

(M,6,,O,null,null)
(O,6,,x,null,null)
(x,6,,v,null,null)
(v,15,,S,null,null)
(S,18,,,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,null,null)
(,,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,D,null,null)
(D,1,,F,null,null)
(F,1,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,j,null,null)
(j,1,,5,null,null)

(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,(),null,null)
((),2,null,-1,(,))
(-1,2,null,4;,-,1)
(4;,2,null,BC,4,;)
(BC,2,null,A,B,C)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,x,null,null)
(x,6,,v,null,null)
(v,15,,S,null,null)
(S,18,,,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,,null,null)
(,,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)

(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)
PRINTING LINKED LIST
(DUMMY,0,,J,null,null)
(J,1,,P,null,null)
(P,1,,W,null,null)
(W,1,,j,null,null)
(j,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,(),null,null)
((),2,null,-1,(,))
(-1,2,null,4,-,1)
(4,,2,null,BC,4,,)
(BC,2,null,DF,B,C)
(DF,2,null,A,D,F)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,x,null,null)
(x,6,,v,null,null)
(v,15,,S,null,null)
(S,18,,,,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,,null,null)
(,,43,,u,null,null)
(u,46,,m,null,null)


```

(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,W,null,null)
(W,1,,j,null,null)
(j,1,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,(),null,null)
((),2,null,-1,(,))
(-1,2,null,4,,-,1)
(4,2,null,BC,4,,:)
(BC,2,null,DF,B,C)
(DF,2,null,JP,D,F)
(JP,2,null,A,J,P)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,x,null,null)
(x,6,,v,null,null)
(v,15,,S,null,null)

```

(S,18,,,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,,null,null)
(.,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,5,null,null)
(5,2,,8,null,null)
(8,2,,G,null,null)
(G,2,,(),null,null)
((),2,null,-1,(,))
(-1,2,null,4;,-,1)
(4;,2,null,BC,4,;)
(BC,2,null,DF,B,C)
(DF,2,null,JP,D,F)
(JP,2,null,Wj,J,P)
(Wj,2,null,A,W,j)

(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,x,null,null)
(x,6,,v,null,null)
(v,15,,S,null,null)
(S,18,,,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,,null,null)
(,,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)
PRINTING LINKED LIST
(DUMMY,0,,G,null,null)
(G,2,,(),null,null)

((),2,null,-1,(,))
(-1,2,null,4,,-,1)
(4,,2,null,BC,4,,)
(BC,2,null,DF,B,C)
(DF,2,null,JP,D,F)
(JP,2,null,Wj,J,P)
(Wj,2,null,A,W,j)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,58,null,null)
(58,4,null,'5,8)
('5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,x,null,null)
(x,6,,v,null,null)
(v,15,,S,null,null)
(S,18,,,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,,null,null)
(,,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)

```
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,, -1,null,null)
(-1,2,null,4;,-,1)
(4;,2,null,BC,4;,)
(BC,2,null,DF,B,C)
(DF,2,null,JP,D,F)
(JP,2,null,Wj,J,P)
(Wj,2,null,A,W,j)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,58,null,null)
(58,4,null,G(),5,8)
(G(),4,null,'G,())
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,x,null,null)
(x,6,,v,null,null)
(v,15,,S,null,null)
(S,18,,,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,,null,null)
(.,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
```

(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,BC,null,null)
(BC,2,null,DF,B,C)
(DF,2,null,JP,D,F)
(JP,2,null,Wj,J,P)
(Wj,2,null,A,W,j)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,58,null,null)
(58,4,null,G(),5,8)
(G(),4,null,-14;,G,())
(-14;,4,null,'-1,4;)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,x,null,null)
(x,6,,v,null,null)
(v,15,,S,null,null)
(S,18,, ,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)

(y,40,,,null,null)
(,,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,JP,null,null)
(JP,2,null,Wj,J,P)
(Wj,2,null,A,W,j)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,58,null,null)
(58,4,null,G(),5,8)
(G(),4,null,-14;,G,())
(-14;,4,null,BCDF,-1,4;)
(BCDF,4,null,',BC,DF)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,x,null,null)
(x,6,,v,null,null)
(v,15,,S,null,null)
(S,18,,,null,null)
(.,27,,w,null,null)

```

(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,null,null)
(,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,A,null,null)
(A,3,,H,null,null)
(H,3,,U,null,null)
(U,3,,58,null,null)
(58,4,null,G(),5,8)
(G(),4,null,-14;,G,())
(-14;,4,null,BCDF,-1,4;)
(BCDF,4,null,JPWj,BC,DF)
(JPWj,4,null,'JP,Wj)
(',5,,T,null,null)
(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,x,null,null)

```



```

(x,6,,v,null,null)
(v,15,,S,null,null)
(S,18,,,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,,null,null)
(.,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,U,null,null)
(U,3,,58,null,null)
(58,4,null,G(),5,8)
(G(),4,null,-14;,G,())
(-14;,4,null,BCDF,-1,4;)
(BCDF,4,null,JPWj,BC,DF)
(JPWj,4,null,',JP,Wj)
(',5,,T,null,null)
(T,5,,M,null,null)

```

```

(M,6,,O,null,null)
(O,6,,x,null,null)
(x,6,,AH,null,null)
(AH,6,null,v,A,H)
(v,15,,S,null,null)
(S,18,,,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,null,null)
(.,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,G(),null,null)
(G(),4,null,-14;,G,())
(-14;,4,null,BCDF,-1,4;)
(BCDF,4,null,JPWj,BC,DF)
(JPWj,4,null,',JP,Wj)
(',5,,T,null,null)

```

```

(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,x,null,null)
(x,6,,AH,null,null)
(AH,6,null,U58,A,H)
(U58,7,null,v,U,58)
(v,15,,S,null,null)
(S,18,,,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,null,null)
(.,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,BCDF,null,null)
(BCDF,4,null,JPWj,BC,DF)
(JPWj,4,null,',JP,Wj)
(',5,,T,null,null)

```

```

(T,5,,M,null,null)
(M,6,,O,null,null)
(O,6,,x,null,null)
(x,6,,AH,null,null)
(AH,6,null,U58,A,H)
(U58,7,null,G()-14;,U,58)
(G()-14;,8,null,v,G(),-14;)
(v,15,,S,null,null)
(S,18,,,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,,null,null)
(.,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,',null,null)
(',5,,T,null,null)
(T,5,,M,null,null)

```

```

(M,6,,O,null,null)
(O,6,,x,null,null)
(x,6,,AH,null,null)
(AH,6,null,U58,A,H)
(U58,7,null,G()-14;,U,58)
(G()-14;,8,null,BCDFJPWj,G(),-14;)
(BCDFJPWj,8,null,v,BCDF,JPWj)
(v,15,,S,null,null)
(S,18,,,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,,null,null)
(.,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,M,null,null)
(M,6,,O,null,null)
(O,6,,x,null,null)

```

```
(x,6,,AH,null,null)
(AH,6,null,U58,A,H)
(U58,7,null,G()-14;,U,58)
(G()-14;,8,null,BCDFJPWj,G(),-14;)
(BCDFJPWj,8,null,'T,BCDF,JPWj)
('T,10,null,v,',T)
(v,15,,S,null,null)
(S,18,,,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,null,null)
(.,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,x,null,null)
(x,6,,AH,null,null)
(AH,6,null,U58,A,H)
(U58,7,null,G()-14;,U,58)
```

```

(G()-14;,8,null,BCDFJPWj,G(),-14;)
(BCDFJPWj,8,null,'T,BCDF,JPWj)
('T,10,null,MO,'T)
(MO,12,null,v,M,O)
(v,15,,S,null,null)
(S,18,,,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,,null,null)
(.,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,U58,null,null)
(U58,7,null,G()-14;,U,58)
(G()-14;,8,null,BCDFJPWj,G(),-14;)
(BCDFJPWj,8,null,'T,BCDF,JPWj)
('T,10,null,MO,'T)
(MO,12,null,xAH,M,O)

```

```

(xAH,12,null,v,x,AH)
(v,15,,S,null,null)
(S,18,,,null,null)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,,null,null)
(.,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,BCDFJPWj,null,null)
(BCDFJPWj,8,null,'T,BCDF,JPWj)
('T,10,null,MO,'T)
(MO,12,null,xAH,M,O)
(xAH,12,null,v,x,AH)
(v,15,,U58G()-14:,null,null)
(U58G()-14:,15,null,S,U58,G()-14;)
(S,18,,,null,null)
(.,27,,w,null,null)

```


(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,null,null)
(,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)

(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,MO,null,null)
(MO,12,null,xAH,M,O)
(xAH,12,null,v,x,AH)
(v,15,,U58G()-14;,null,null)
(U58G()-14;,15,null,S,U58,G()-14;)
(S,18,,BCDFJPWj'T,null,null)
(BCDFJPWj'T,18,null,,BCDFJPWj,'T)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)

(y,40,,,null,null)
(,,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,v,null,null)
(v,15,,U58G()-14;,null,null)
(U58G()-14;,15,null,S,U58,G()-14;)
(S,18,,BCDFJPWj'T,null,null)
(BCDFJPWj'T,18,null,MOxAH,BCDFJPWj,'T)
(MOxAH,24,null,,MO,xAH)
(.,27,,w,null,null)
(w,30,,k,null,null)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,null,null)
(,,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)

(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,S,null,null)
(S,18,,BCDFJPWj'T,null,null)
(BCDFJPWj'T,18,null,MOxAH,BCDFJPWj,'T)
(MOxAH,24,null,,MO,xAH)
(.,27,,w,null,null)
(w,30,,vU58G()-14;,null,null)
(vU58G()-14;,30,null,k,v,U58G()-14;)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,,null,null)
(,,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)

```

(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,MOxAH,null,null)
(MOxAH,24,null,,MO,xAH)
(.,27,,w,null,null)
(w,30,,vU58G()-14;,null,null)
(vU58G()-14;,30,null,SBCDFJPWj'T,v,U58G()-14;)
(SBCDFJPWj'T,36,null,k,S,BCDFJPWj'T)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,,null,null)
(.,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,g,null,null)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,w,null,null)
(w,30,,vU58G()-14;,null,null)

```

(vU58G()-14;,30,null,SBCDFJPWj'T,v,U58G()-14;)
(SBCDFJPWj'T,36,null,k,S,BCDFJPWj'T)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,,null,null)
(,,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,MOxAH.,null,null)
(MOxAH.,51,null,g,MOxAH.,.)
(g,65,,f,null,null)
(f,71,,d,null,null)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,SBCDFJPWj'T,null,null)
(SBCDFJPWj'T,36,null,k,S,BCDFJPWj'T)
(k,37,,c,null,null)
(c,38,,b,null,null)
(b,40,,p,null,null)
(p,40,,y,null,null)
(y,40,,,,null,null)
(,,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,MOxAH.,null,null)
(MOxAH.,51,null,wvU58G()-14;,MOxAH.,.)

(wvU58G()-14;,60,null,g,w,vU58G()-14;)

(g,65,,f,null,null)

(f,71,,d,null,null)

(d,84,,l,null,null)

(l,101,,r,null,null)

(r,126,,o,null,null)

(o,171,,h,null,null)

(h,184,,s,null,null)

(s,191,,i,null,null)

(i,196,,n,null,null)

(n,199,,t,null,null)

(t,220,,a,null,null)

(a,241,,e,null,null)

(e,270,, ,null,null)

(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,c,null,null)

(c,38,,b,null,null)

(b,40,,p,null,null)

(p,40,,y,null,null)

(y,40,,,null,null)

(,43,,u,null,null)

(u,46,,m,null,null)

(m,49,,MOxAH.,null,null)

(MOxAH.,51,null,wvU58G()-14;,MOxAH.,)

(wvU58G()-14;,60,null,g,w,vU58G()-14;)

(g,65,,f,null,null)

(f,71,,SBCDFJPWj'Tk,null,null)

(SBCDFJPWj'Tk,73,null,d,SBCDFJPWj'T,k)

(d,84,,l,null,null)

(l,101,,r,null,null)

(r,126,,o,null,null)

(o,171,,h,null,null)

(h,184,,s,null,null)

(s,191,,i,null,null)

(i,196,,n,null,null)

(n,199,,t,null,null)

```

(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,p,null,null)
(p,40,,y,null,null)
(y,40,,,null,null)
(,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,MOxAH.,null,null)
(MOxAH.,51,null,wvU58G()-14;,MOxAH.,)
(wvU58G()-14;,60,null,g,w,vU58G()-14;)
(g,65,,f,null,null)
(f,71,,SBCDFJPWj'Tk,null,null)
(SBCDFJPWj'Tk,73,null,cb,SBCDFJPWj'T,k)
(cb,78,null,d,c,b)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,,null,null)
(,43,,u,null,null)
(u,46,,m,null,null)
(m,49,,MOxAH.,null,null)
(MOxAH.,51,null,wvU58G()-14;,MOxAH.,)
(wvU58G()-14;,60,null,g,w,vU58G()-14;)
(g,65,,f,null,null)

```

```

(f,71,,SBCDFJPWj'Tk,null,null)
(SBCDFJPWj'Tk,73,null,cb,SBCDFJPWj'T,k)
(cb,78,null,py,c,b)
(py,80,null,d,p,y)
(d,84,,l,null,null)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,m,null,null)
(m,49,,MOxAH.,null,null)
(MOxAH.,51,null,wvU58G()-14;,MOxAH.,)
(wvU58G()-14;,60,null,g,w,vU58G()-14;)
(g,65,,f,null,null)
(f,71,,SBCDFJPWj'Tk,null,null)
(SBCDFJPWj'Tk,73,null,cb,SBCDFJPWj'T,k)
(cb,78,null,py,c,b)
(py,80,null,d,p,y)
(d,84,,,u,null,null)
(u,89,null,l,,,u)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)

```



```

(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,wvU58G()-14;,null,null)
(wvU58G()-14;,60,null,g,w,vU58G()-14;)
(g,65,,f,null,null)
(f,71,,SBCDFJPWj'Tk,null,null)
(SBCDFJPWj'Tk,73,null,cb,SBCDFJPWj'T,k)
(cb,78,null,py,c,b)
(py,80,null,d,p,y)
(d,84,,,u,null,null)
(u,89,null,mMOxAH.,,,u)
(mMOxAH.,100,null,l,m,MOxAH.)
(l,101,,r,null,null)
(r,126,,o,null,null)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,f,null,null)
(f,71,,SBCDFJPWj'Tk,null,null)
(SBCDFJPWj'Tk,73,null,cb,SBCDFJPWj'T,k)
(cb,78,null,py,c,b)
(py,80,null,d,p,y)
(d,84,,,u,null,null)
(u,89,null,mMOxAH.,,,u)
(mMOxAH.,100,null,l,m,MOxAH.)
(l,101,,wvU58G()-14;g,null,null)
(wvU58G()-14;g,125,null,r,wvU58G()-14;,g)
(r,126,,o,null,null)
(o,171,,h,null,null)

```

```

(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,cb,null,null)
(cb,78,null,py,c,b)
(py,80,null,d,p,y)
(d,84,,,u,null,null)
(u,89,null,mMOxAH,,,u)
(mMOxAH.,100,null,l,m,MOxAH.)
(l,101,,wvU58G()-14;g,null,null)
(wvU58G()-14;g,125,null,r,wvU58G()-14;;g)
(r,126,,fSBCDFJPWj'Tk,null,null)
(fSBCDFJPWj'Tk,144,null,o,f,SBCDFJPWj'Tk)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,d,null,null)
(d,84,,,u,null,null)
(u,89,null,mMOxAH,,,u)
(mMOxAH.,100,null,l,m,MOxAH.)
(l,101,,wvU58G()-14;g,null,null)
(wvU58G()-14;g,125,null,r,wvU58G()-14;;g)
(r,126,,fSBCDFJPWj'Tk,null,null)
(fSBCDFJPWj'Tk,144,null,cbpy,f,SBCDFJPWj'Tk)

```

(cbpy,158,null,o,cb,py)
(o,171,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,mMOxAH.,null,null)
(mMOxAH.,100,null,l,m,MOxAH.)
(l,101,,wvU58G()-14;g,null,null)
(wvU58G()-14;g,125,null,r,wvU58G()-14;;g)
(r,126,,fSBCDFJPWj'Tk,null,null)
(fSBCDFJPWj'Tk,144,null,cbpy,f,SBCDFJPWj'Tk)
(cbpy,158,null,o,cb,py)
(o,171,,d,u,null,null)
(d,u,173,null,h,d,,u)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,t,null,null)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,wvU58G()-14;g,null,null)
(wvU58G()-14;g,125,null,r,wvU58G()-14;;g)
(r,126,,fSBCDFJPWj'Tk,null,null)
(fSBCDFJPWj'Tk,144,null,cbpy,f,SBCDFJPWj'Tk)
(cbpy,158,null,o,cb,py)
(o,171,,d,u,null,null)
(d,u,173,null,h,d,,u)
(h,184,,s,null,null)

```

(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,mMOxAH.l,null,null)
(mMOxAH.l,201,null,t,mMOxAH.,l)
(t,220,,a,null,null)
(a,241,,e,null,null)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,fSBCDFJPWj'Tk,null,null)
(fSBCDFJPWj'Tk,144,null,cbpy,f,SBCDFJPWj'Tk)
(cbpy,158,null,o,cb,py)
(o,171,,d,u,null,null)
(d,u,173,null,h,d,,u)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,mMOxAH.l,null,null)
(mMOxAH.l,201,null,t,mMOxAH.,l)
(t,220,,a,null,null)
(a,241,,wvU58G()-14;gr,null,null)
(wvU58G()-14;gr,251,null,e,wvU58G()-14;g,r)
(e,270,, ,null,null)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,o,null,null)
(o,171,,d,u,null,null)
(d,u,173,null,h,d,,u)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,mMOxAH.l,null,null)
(mMOxAH.l,201,null,t,mMOxAH.,l)
(t,220,,a,null,null)
(a,241,,wvU58G()-14;gr,null,null)
(wvU58G()-14;gr,251,null,e,wvU58G()-14;g,r)
(e,270,,fSBCDFJPWj'Tkcbpy,null,null)

```

```

(fSBCDFJPWj'Tkcbpy,302,null, ,fSBCDFJPWj'Tk,cbpy)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,h,null,null)
(h,184,,s,null,null)
(s,191,,i,null,null)
(i,196,,n,null,null)
(n,199,,mMOxAH.l,null,null)
(mMOxAH.l,201,null,t,mMOxAH.,l)
(t,220,,a,null,null)
(a,241,,wvU58G()-14;gr,null,null)
(wvU58G()-14;gr,251,null,e,wvU58G()-14;g,r)
(e,270,,fSBCDFJPWj'Tkcbpy,null,null)
(fSBCDFJPWj'Tkcbpy,302,null,od,u,fSBCDFJPWj'Tk,cbpy)
(od,u,344,null, ,o,d,u)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,i,null,null)
(i,196,,n,null,null)
(n,199,,mMOxAH.l,null,null)
(mMOxAH.l,201,null,t,mMOxAH.,l)
(t,220,,a,null,null)
(a,241,,wvU58G()-14;gr,null,null)
(wvU58G()-14;gr,251,null,e,wvU58G()-14;g,r)
(e,270,,fSBCDFJPWj'Tkcbpy,null,null)
(fSBCDFJPWj'Tkcbpy,302,null,od,u,fSBCDFJPWj'Tk,cbpy)
(od,u,344,null,hs,o,d,u)
(hs,375,null, ,h,s)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,mMOxAH.l,null,null)
(mMOxAH.l,201,null,t,mMOxAH.,l)
(t,220,,a,null,null)
(a,241,,wvU58G()-14;gr,null,null)
(wvU58G()-14;gr,251,null,e,wvU58G()-14;g,r)
(e,270,,fSBCDFJPWj'Tkcbpy,null,null)
(fSBCDFJPWj'Tkcbpy,302,null,od,u,fSBCDFJPWj'Tk,cbpy)

```

```

(od,u,344,null,hs,o,d,u)
(hs,375,null,in,h,s)
(in,395,null, ,i,n)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,a,null,null)
(a,241,,wvU58G()-14;gr,null,null)
(wvU58G()-14;gr,251,null,e,wvU58G()-14;g,r)
(e,270,,fSBCDFJPWj'Tkcbpy,null,null)
(fSBCDFJPWj'Tkcbpy,302,null,od,u,fSBCDFJPWj'Tk,cbpy)
(od,u,344,null,hs,o,d,u)
(hs,375,null,in,h,s)
(in,395,null,mMOxAH.lt,i,n)
(mMOxAH.lt,421,null, ,mMOxAH.l,t)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,e,null,null)
(e,270,,fSBCDFJPWj'Tkcbpy,null,null)
(fSBCDFJPWj'Tkcbpy,302,null,od,u,fSBCDFJPWj'Tk,cbpy)
(od,u,344,null,hs,o,d,u)
(hs,375,null,in,h,s)
(in,395,null,mMOxAH.lt,i,n)
(mMOxAH.lt,421,null,awvU58G()-14;gr,mMOxAH.l,t)
(awvU58G()-14;gr,492,null, ,a,wvU58G()-14;gr)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,od,u,null,null)
(od,u,344,null,hs,o,d,u)
(hs,375,null,in,h,s)
(in,395,null,mMOxAH.lt,i,n)
(mMOxAH.lt,421,null,awvU58G()-14;gr,mMOxAH.l,t)
(awvU58G()-14;gr,492,null,efSBCDFJPWj'Tkcbpy,a,wvU58G()-14;gr)
(efSBCDFJPWj'Tkcbpy,572,null, ,e,fSBCDFJPWj'Tkcbpy)
( ,578,,NULL,NULL,NULL)
***PRINTING LINKED LIST***
(DUMMY,0,,in,null,null)
(in,395,null,mMOxAH.lt,i,n)

```

(mMOxAH.lt,421,null,awvU58G()-14;gr,mMOxAH.l,t)
(awvU58G()-14;gr,492,null,efSBCDFJPWj'Tkcbpy,a,wvU58G()-14;gr)
(efSBCDFJPWj'Tkcbpy,572,null, ,e,fSBCDFJPWj'Tkcbpy)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,,awvU58G()-14;gr,null,null)
(awvU58G()-14;gr,492,null,efSBCDFJPWj'Tkcbpy,a,wvU58G()-14;gr)
(efSBCDFJPWj'Tkcbpy,572,null, ,e,fSBCDFJPWj'Tkcbpy)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

(DUMMY,0,, ,null,null)
(,578,,NULL,NULL,NULL)

PRINTING LINKED LIST

PRINTING LINKED LIST

(DUMMY,0,, ,null,null)
(,578,,NULL,NULL,NULL)

PREORDER TREE TRAVERSAL

(awvU58G()-14;grefSBCDFJPWj'Tkcbpy,1064,null,null,awvU58G()-14;gr,efSBCDFJPWj'Tkcbpy)
(awvU58G()-14;gr,492,null,efSBCDFJPWj'Tkcbpy,a,wvU58G()-14;gr)
(a,241,00,wvU58G()-14;gr,null,null)
(wvU58G()-14;gr,251,null,e,wvU58G()-14;g,r)
(wvU58G()-14;g,125,null,r,wvU58G()-14;,g)
(wvU58G()-14;,60,null,g,w,vU58G()-14;)
(w,30,01000,vU58G()-14;,null,null)
(vU58G()-14;,30,null,SBCDFJPWj'T,v,U58G()-14;)
(v,15,010010,U58G()-14;,null,null)
(U58G()-14;,15,null,S,U58,G()-14;)
(U58,7,null,G()-14;,U,58)
(U,3,01001100,58,null,null)
(58,4,null,G(),5,8)
(5,2,010011010,8,null,null)
(8,2,010011011,G,null,null)
(G()-14;,8,null,BCDFJPWj,G(),-14;)
(G(),4,null,-14;,G,())
(G,2,010011100,(),null,null)

((,2,null,-1,(,))
(,1,0100111010,),null,null)
(,1,0100111011,-,null,null)
(-14;,4,null,BCDF,-1,4;)
(-1,2,null,4;,-,1)
(-,1,0100111100,1,null,null)
(1,1,0100111101,4,null,null)
(4;,2,null,BC,4,;)
(4,1,0100111110,;,null,null)
(;,1,0100111111,B,null,null)
(g,65,0101,f,null,null)
(r,126,011,fSBCDFJPWj'Tk,null,null)
(efSBCDFJPWj'Tkcbpy,572,null, ,e,fSBCDFJPWj'Tkcbpy)
(e,270,10,fSBCDFJPWj'Tkcbpy,null,null)
(fSBCDFJPWj'Tkcbpy,302,null,od,u,fSBCDFJPWj'Tk,cbpy)
(fSBCDFJPWj'Tk,144,null,cbpy,f,SBCDFJPWj'Tk)
(f,71,1100,SBCDFJPWj'Tk,null,null)
(SBCDFJPWj'Tk,73,null,cb,SBCDFJPWj'T,k)
(SBCDFJPWj'T,36,null,k,S,BCDFJPWj'T)
(S,18,110100,BCDFJPWj'T,null,null)
(BCDFJPWj'T,18,null,MOxAH,BCDFJPWj,'T)
(BCDFJPWj,8,null,'T,BCDF,JPWj)
(BCDF,4,null,JPWj,BC,DF)
(BC,2,null,DF,B,C)
(B,1,1101010000,C,null,null)
(C,1,1101010001,D,null,null)
(DF,2,null,JP,D,F)
(D,1,1101010010,F,null,null)
(F,1,1101010011,J,null,null)
(JPWj,4,null,',JP,Wj)
(JP,2,null,Wj,J,P)
(J,1,1101010100,P,null,null)
(P,1,1101010101,W,null,null)
(Wj,2,null,A,W,j)
(W,1,1101010110,j,null,null)
(j,1,1101010111,5,null,null)
(T,10,null,MO,',T)

('5,11010110,T,null,null)
 (T,5,11010111,M,null,null)
 (k,37,11011,c,null,null)
 (cbpy,158,null,o,cb,py)
 (cb,78,null,py,c,b)
 (c,38,11100,b,null,null)
 (b,40,11101,p,null,null)
 (py,80,null,d,p,y)
 (p,40,11110,y,null,null)
 (y,40,11111,,,null,null)

POSTORDER TREE TRAVERSAL

(awvU58G()-14;gr,492,null,efSBCDFJPWj'Tkcbpy,a,wvU58G()-14;gr)
 (a,241,00,wvU58G()-14;gr,null,null)
 (wvU58G()-14;gr,251,null,e,wvU58G()-14;g,r)
 (wvU58G()-14;g,125,null,r,wvU58G()-14;,g)
 (wvU58G()-14;,60,null,g,w,vU58G()-14;)
 (w,30,01000,vU58G()-14;,null,null)
 (vU58G()-14;,30,null,SBCDFJPWj'T,v,U58G()-14;)
 (v,15,010010,U58G()-14;,null,null)
 (U58G()-14;,15,null,S,U58,G()-14;)
 (U58,7,null,G()-14;,U,58)
 (U,3,01001100,58,null,null)
 (58,4,null,G(),5,8)
 (5,2,010011010,8,null,null)
 (8,2,010011011,G,null,null)
 (G()-14;,8,null,BCDFJPWj,G(),-14;)
 (G(),4,null,-14;,G,())
 (G,2,010011100,(),null,null)
 ((),2,null,-1,(,))
 ((,1,0100111010,),null,null)
 (,1,0100111011,-,null,null)
 (-14;,4,null,BCDF,-1,4;)
 (-1,2,null,4;,-,1)
 (-,1,0100111100,1,null,null)
 (1,1,0100111101,4,null,null)
 (4;,2,null,BC,4;,)

(4,1,0100111110,,null,null)
(;,1,0100111111,B,null,null)
(g,65,0101,f,null,null)
(r,126,011,fSBCDFJPWj'Tk,null,null)
(efSBCDFJPWj'Tkcbpy,572,null, ,e,fSBCDFJPWj'Tkcbpy)
(e,270,10,fSBCDFJPWj'Tkcbpy,null,null)
(fSBCDFJPWj'Tkcbpy,302,null,od,u,fSBCDFJPWj'Tk,cbpy)
(fSBCDFJPWj'Tk,144,null,cbpy,f,SBCDFJPWj'Tk)
(f,71,1100,SBCDFJPWj'Tk,null,null)
(SBCDFJPWj'Tk,73,null,cb,SBCDFJPWj'T,k)
(SBCDFJPWj'T,36,null,k,S,BCDFJPWj'T)
(S,18,110100,BCDFJPWj'T,null,null)
(BCDFJPWj'T,18,null,MOxAH,BCDFJPWj,'T)
(BCDFJPWj,8,null,'T,BCDF,JPWj)
(BCDF,4,null,JPWj,BC,DF)
(BC,2,null,DF,B,C)
(B,1,1101010000,C,null,null)
(C,1,1101010001,D,null,null)
(DF,2,null,JP,D,F)
(D,1,1101010010,F,null,null)
(F,1,1101010011,J,null,null)
(JPWj,4,null,',JP,Wj)
(JP,2,null,Wj,J,P)
(J,1,1101010100,P,null,null)
(P,1,1101010101,W,null,null)
(Wj,2,null,A,W,j)
(W,1,1101010110,j,null,null)
(j,1,1101010111,5,null,null)
(T,10,null,MO,',T)
(',5,11010110,T,null,null)
(T,5,11010111,M,null,null)
(k,37,11011,c,null,null)
(cbpy,158,null,o,cb,py)
(cb,78,null,py,c,b)
(c,38,11100,b,null,null)
(b,40,11101,p,null,null)
(py,80,null,d,p,y)

(p,40,11110,y,null,null)
(y,40,11111,,,null,null)
(awvU58G()-14;grefSBCDFJPWj'Tkcbpy,1064,null,null,awvU58G()-14;gr,efSBCDFJPWj'Tkcbpy)

INORDER TREE TRAVERSAL

(awvU58G()-14;gr,492,null,efSBCDFJPWj'Tkcbpy,a,wvU58G()-14;gr)
(a,241,00,wvU58G()-14;gr,null,null)
(wvU58G()-14;gr,251,null,e,wvU58G()-14;g,r)
(wvU58G()-14;g,125,null,r,wvU58G()-14;,g)
(wvU58G()-14;,60,null,g,w,vU58G()-14;)
(w,30,01000,vU58G()-14;,null,null)
(vU58G()-14;,30,null,SBCDFJPWj'T,v,U58G()-14;)
(v,15,010010,U58G()-14;,null,null)
(U58G()-14;,15,null,S,U58,G()-14;)
(U58,7,null,G()-14;,U,58)
(U,3,01001100,58,null,null)
(58,4,null,G(),5,8)
(5,2,010011010,8,null,null)
(8,2,010011011,G,null,null)
(G()-14;,8,null,BCDFJPWj,G(),-14;)
(G(),4,null,-14;,G(),)
(G,2,010011100,(),null,null)
((),2,null,-1,(,))
((),1,0100111010,(),null,null)
((),1,0100111011,-,null,null)
(-14;,4,null,BCDF,-1,4;)
(-1,2,null,4;,-,1)
(-,1,0100111100,1,null,null)
(1,1,0100111101,4,null,null)
(4;,2,null,BC,4;,)
(4,1,0100111110,;,null,null)
(;,1,0100111111,B,null,null)
(g,65,0101,f,null,null)
(r,126,011,fSBCDFJPWj'Tk,null,null)
(awvU58G()-14;grefSBCDFJPWj'Tkcbpy,1064,null,null,awvU58G()-14;gr,efSBCDFJPWj'Tkcbpy)
(efSBCDFJPWj'Tkcbpy,572,null, ,e,fSBCDFJPWj'Tkcbpy)
(e,270,10,fSBCDFJPWj'Tkcbpy,null,null)

(fSBCDFJPWj'Tkcbpy,302,null,od,u,fSBCDFJPWj'Tk,cbpy)
(fSBCDFJPWj'Tk,144,null,cbpy,f,SBCDFJPWj'Tk)
(f,71,1100,SBCDFJPWj'Tk,null,null)
(SBCDFJPWj'Tk,73,null,cb,SBCDFJPWj'T,k)
(SBCDFJPWj'T,36,null,k,S,BCDFJPWj'T)
(S,18,110100,BCDFJPWj'T,null,null)
(BCDFJPWj'T,18,null,MOxAH,BCDFJPWj,'T)
(BCDFJPWj,8,null,'T,BCDF,JPWj)
(BCDF,4,null,JPWj,BC,DF)
(BC,2,null,DF,B,C)
(B,1,1101010000,C,null,null)
(C,1,1101010001,D,null,null)
(DF,2,null,JP,D,F)
(D,1,1101010010,F,null,null)
(F,1,1101010011,J,null,null)
(JPWj,4,null,'JP,Wj)
(JP,2,null,Wj,J,P)
(J,1,1101010100,P,null,null)
(P,1,1101010101,W,null,null)
(Wj,2,null,A,W,j)
(W,1,1101010110,j,null,null)
(j,1,1101010111,5,null,null)
(T,10,null,MO,'T)
(',5,11010110,T,null,null)
(T,5,11010111,M,null,null)
(k,37,11011,c,null,null)
(cbpy,158,null,o,cb,py)
(cb,78,null,py,c,b)
(c,38,11100,b,null,null)
(b,40,11101,p,null,null)
(py,80,null,d,p,y)
(p,40,11110,y,null,null)
(y,40,11111,,,null,null)
char:65279
char:84 11010111
char:101 10
char:97 00

char:97 00
char:101 10
char:83 110100
char:101 10
char:97 00
char:101 10
char:101 10
char:114 011
char:121 11111
char:102 1100
char:97 00
char:98 11101
char:97 00
char:101 10
char:98 11101
char:101 10
char:119 01000
char:101 10
char:101 10
char:97 00
char:97 00
char:103 0101
char:103 0101
char:101 10
char:112 11110
char:101 10
char:114 011
char:101 10
char:99 11100
char:101 10
char:102 1100
char:101 10
char:114 011
char:97 00
char:83 110100
char:97 00
char:97 00

char:103 0101
char:97 00
char:97 00
char:97 00
char:114 011
char:103 0101
char:101 10
char:97 00
char:114 011
char:84 11010111
char:101 10
char:114 011
char:121 11111
char:112 11110
char:101 10
char:119 01000
char:83 110100
char:97 00
char:97 00
char:103 0101
char:97 00
char:118 010010
char:103 0101
char:103 0101
char:101 10
char:56 010011011
char:52 0100111110
char:97 00
char:121 11111
char:119 01000
char:99 11100
char:97 00
char:99 11100
char:103 0101
char:97 00
char:102 1100
char:97 00

char:119 01000
char:98 11101
char:101 10
char:103 0101
char:101 10
char:101 10
char:97 00
char:97 00
char:97 00
char:101 10
char:119 01000
char:114 011
char:102 1100
char:114 011
char:102 1100
char:99 11100
char:107 11011
char:101 10
char:101 10
char:99 11100
char:107 11011
char:121 11111
char:97 00
char:121 11111
char:103 0101
char:97 00
char:112 11110
char:112 11110
char:114 011
char:101 10
char:99 11100
char:101 10
char:97 00
char:97 00
char:98 11101
char:101 10
char:101 10

char:102 1100
char:114 011
char:98 11101
char:101 10
char:98 11101
char:121 11111
char:112 11110
char:97 00
char:114 011
char:101 10
char:97 00
char:119 01000
char:97 00
char:97 00
char:98 11101
char:101 10
char:101 10
char:101 10
char:97 00
char:102 1100
char:119 01000
char:99 11100
char:99 11100
char:101 10
char:102 1100
char:102 1100
char:101 10
char:114 011
char:101 10
char:84 11010111
char:101 10
char:98 11101
char:121 11111
char:118 010010
char:83 110100
char:97 00
char:97 00

char:103 0101
char:39 11010110
char:97 00
char:99 11100
char:107 11011
char:101 10
char:97 00
char:99 11100
char:103 0101
char:97 00
char:103 0101
char:102 1100
char:103 0101
char:103 0101
char:101 10
char:97 00
char:114 011
char:112 11110
char:114 011
char:101 10
char:112 11110
char:97 00
char:114 011
char:103 0101
char:102 1100
char:97 00
char:107 11011
char:103 0101
char:97 00
char:98 11101
char:101 10
char:114 011
char:99 11100
char:97 00
char:98 11101
char:97 00
char:101 10

char:98 11101
char:97 00
char:97 00
char:102 1100
char:97 00
char:118 010010
char:114 011
char:101 10
char:112 11110
char:97 00
char:121 11111
char:101 10
char:114 011
char:74 1101010100
char:101 10
char:68 1101010010
char:97 00
char:103 0101
char:103 0101
char:83 110100
char:97 00
char:97 00
char:103 0101
char:101 10
char:97 00
char:97 00
char:101 10
char:101 10
char:97 00
char:121 11111
char:101 10
char:119 01000
char:118 010010
char:101 10
char:114 011
char:101 10
char:102 1100

char:97 00
char:114 011
char:101 10
char:71 010011100
char:102 1100
char:83 110100
char:114 011
char:101 10
char:97 00
char:114 011
char:102 1100
char:67 1101010001
char:98 11101
char:97 00
char:101 10
char:83 110100
char:114 011
char:97 00
char:102 1100
char:70 1101010011
char:114 011
char:97 00
char:102 1100
char:99 11100
char:102 1100
char:101 10
char:97 00
char:99 11100
char:107 11011
char:121 11111
char:114 011
char:101 10
char:97 00
char:107 11011
char:101 10
char:97 00
char:114 011

char:101 10
char:101 10
char:101 10
char:103 0101
char:121 11111
char:45 0100111100
char:102 1100
char:102 1100
char:97 00
char:121 11111
char:102 1100
char:99 11100
char:107 11011
char:121 11111
char:114 011
char:101 10
char:97 00
char:107 11011
char:83 110100
char:97 00
char:97 00
char:103 0101
char:97 00
char:107 11011
char:101 10
char:107 11011
char:102 1100
char:102 1100
char:101 10
char:71 010011100
char:102 1100
char:83 110100
char:114 011
char:101 10
char:97 00
char:101 10
char:101 10

char:97 00
char:98 11101
char:121 11111
char:97 00
char:98 11101
char:97 00
char:97 00
char:107 11011
char:101 10
char:98 11101
char:121 11111
char:97 00
char:98 11101
char:103 0101
char:102 1100
char:97 00
char:101 10
char:114 011
char:101 10
char:97 00
char:97 00
char:114 011
char:85 01001100
char:97 00
char:98 11101
char:101 10
char:97 00
char:101 10
char:103 0101
char:114 011
char:101 10
char:97 00
char:97 00
char:114 011
char:83 110100
char:97 00
char:97 00

char:103 0101
char:101 10
char:97 00
char:112 11110
char:101 10
char:98 11101
char:121 11111
char:101 10
char:97 00
char:114 011
char:97 00
char:119 01000
char:97 00
char:121 11111
char:97 00
char:103 0101
char:112 11110
char:97 00
char:119 01000
char:83 110100
char:97 00
char:97 00
char:103 0101
char:103 0101
char:101 10
char:101 10
char:84 11010111
char:103 0101
char:119 01000
char:101 10
char:98 11101
char:121 11111
char:101 10
char:114 011
char:103 0101
char:103 0101
char:101 10

char:97 00
char:112 11110
char:97 00
char:83 110100
char:97 00
char:97 00
char:103 0101
char:101 10
char:112 11110
char:114 011
char:101 10
char:101 10
char:97 00
char:99 11100
char:112 11110
char:97 00
char:97 00
char:101 10
char:97 00
char:112 11110
char:112 11110
char:114 011
char:101 10
char:99 11100
char:97 00
char:102 1100
char:114 011
char:97 00
char:118 010010
char:101 10
char:114 011
char:97 00
char:114 011
char:121 11111
char:102 1100
char:101 10
char:114 011

char:101 10
char:102 1100
char:101 10
char:114 011
char:114 011
char:103 0101
char:97 00
char:97 00
char:98 11101
char:114 011
char:101 10
char:114 011
char:101 10
char:97 00
char:101 10
char:101 10
char:114 011
char:101 10
char:97 00
char:98 11101
char:101 10
char:99 11100
char:97 00
char:101 10
char:102 1100
char:101 10
char:102 1100
char:39 11010110
char:103 0101
char:114 011
char:101 10
char:97 00
char:103 0101
char:121 11111
char:101 10
char:97 00
char:101 10

char:101 10
char:114 011
char:118 010010
char:101 10
char:101 10
char:97 00
char:101 10
char:97 00
char:114 011
char:101 10
char:114 011
char:97 00
char:121 11111
char:101 10
char:102 1100
char:98 11101
char:101 10
char:103 0101
char:99 11100
char:114 011
char:99 11100
char:101 10
char:101 10
char:107 11011
char:102 1100
char:102 1100
char:83 110100
char:97 00
char:97 00
char:103 0101
char:119 01000
char:114 011
char:97 00
char:97 00
char:101 10
char:114 011
char:101 10

char:97 00
char:114 011
char:101 10
char:97 00
char:103 0101
char:114 011
char:101 10
char:103 0101
char:112 11110
char:101 10
char:102 1100
char:101 10
char:97 00
char:97 00
char:98 11101
char:101 10
char:97 00
char:114 011
char:119 01000
char:97 00
char:97 00
char:114 011
char:112 11110
char:83 110100
char:97 00
char:97 00
char:103 0101
char:114 011
char:97 00
char:112 11110
char:101 10
char:97 00
char:114 011
char:101 10
char:101 10
char:102 1100
char:107 11011

char:102 1100
char:102 1100
char:97 00
char:101 10
char:97 00
char:101 10
char:107 11011
char:103 0101
char:97 00
char:98 11101
char:101 10
char:103 0101
char:112 11110
char:114 011
char:99 11100
char:101 10
char:101 10
char:102 1100
char:119 01000
char:98 11101
char:114 011
char:103 0101
char:97 00
char:101 10
char:97 00
char:114 011
char:107 11011
char:101 10
char:97 00
char:119 01000
char:97 00
char:121 11111
char:112 11110
char:101 10
char:112 11110
char:101 10
char:101 10

char:119 01000
char:102 1100
char:101 10
char:101 10
char:119 01000
char:97 00
char:121 11111
char:114 011
char:101 10
char:97 00
char:114 011
char:107 11011
char:97 00
char:114 011
char:101 10
char:97 00
char:114 011
char:97 00
char:99 11100
char:101 10
char:101 10
char:97 00
char:114 011
char:39 11010110
char:98 11101
char:83 110100
char:97 00
char:97 00
char:103 0101
char:107 11011
char:97 00
char:103 0101
char:114 011
char:101 10
char:97 00
char:97 00
char:107 11011

char:97 00
char:114 011
char:107 11011
char:119 01000
char:97 00
char:114 011
char:112 11110
char:98 11101
char:101 10
char:101 10
char:101 10
char:119 01000
char:101 10
char:97 00
char:112 11110
char:101 10
char:97 00
char:107 11011
char:101 10
char:97 00
char:101 10
char:119 01000
char:97 00
char:114 011
char:112 11110
char:98 11101
char:121 11111
char:114 011
char:97 00
char:112 11110
char:112 11110
char:103 0101
char:107 11011
char:102 1100
char:101 10
char:101 10
char:101 10

char:102 1100
char:97 00
char:97 00
char:114 011
char:101 10
char:112 11110
char:119 01000
char:97 00
char:114 011
char:102 1100
char:102 1100
char:101 10
char:101 10
char:101 10
char:102 1100
char:97 00
char:114 011
char:107 11011
char:59 0100111111
char:102 1100
char:118 010010
char:101 10
char:97 00
char:114 011
char:107 11011
char:97 00
char:114 011
char:101 10
char:97 00
char:97 00
char:97 00
char:121 11111
char:101 10
char:114 011
char:97 00
char:114 011
char:101 10

char:114 011
char:118 010010
char:101 10
char:97 00
char:119 01000
char:97 00
char:121 11111
char:66 1101010000
char:101 10
char:97 00
char:114 011
char:107 11011
char:107 11011
char:101 10
char:101 10
char:112 11110
char:99 11100
char:103 0101
char:97 00
char:98 11101
char:121 11111
char:103 0101
char:102 1100
char:97 00
char:101 10
char:97 00
char:114 011
char:107 11011
char:97 00
char:118 010010
char:101 10
char:97 00
char:101 10
char:118 010010
char:114 011
char:101 10
char:101 10

char:97 00
char:114 011
char:39 11010110
char:101 10
char:114 011
char:101 10
char:99 11100
char:97 00
char:114 011
char:99 11100
char:97 00
char:101 10
char:97 00
char:118 010010
char:103 0101
char:97 00
char:107 11011
char:101 10
char:101 10
char:99 11100
char:103 0101
char:121 11111
char:102 1100
char:98 11101
char:97 00
char:99 11100
char:107 11011
char:98 11101
char:101 10
char:97 00
char:97 00
char:101 10
char:97 00
char:83 110100
char:97 00
char:97 00
char:103 0101

char:107 11011
char:119 01000
char:97 00
char:101 10
char:101 10
char:114 011
char:121 11111
char:101 10
char:97 00
char:101 10
char:101 10
char:97 00
char:114 011
char:107 11011
char:102 1100
char:119 01000
char:101 10
char:121 11111
char:97 00
char:118 010010
char:101 10
char:107 11011
char:101 10
char:114 011
char:101 10
char:97 00
char:85 01001100
char:112 11110
char:114 011
char:101 10
char:97 00
char:99 11100
char:103 0101
char:101 10
char:114 011
char:101 10
char:98 11101

char:101 10
char:102 1100
char:114 011
char:101 10
char:97 00
char:119 01000
char:101 10
char:101 10
char:97 00
char:121 11111
char:83 110100
char:97 00
char:97 00
char:103 0101
char:114 011
char:103 0101
char:103 0101
char:101 10
char:97 00
char:99 11100
char:107 11011
char:99 11100
char:97 00
char:114 011
char:114 011
char:121 11111
char:103 0101
char:101 10
char:101 10
char:97 00
char:118 010010
char:121 11111
char:97 00
char:101 10
char:114 011
char:101 10
char:97 00

char:118 010010
char:103 0101
char:101 10
char:102 1100
char:101 10
char:97 00
char:97 00
char:101 10
char:98 11101
char:101 10
char:101 10
char:114 011
char:101 10
char:99 11100
char:101 10
char:101 10
char:101 10
char:112 11110
char:98 11101
char:101 10
char:97 00
char:102 1100
char:97 00
char:97 00
char:101 10
char:101 10
char:112 11110
char:101 10
char:101 10
char:112 11110
char:103 0101
char:114 011
char:112 11110
char:102 1100
char:102 1100
char:101 10
char:114 011

char:101 10
char:103 0101
char:97 00
char:101 10
char:114 011
char:101 10
char:101 10
char:97 00
char:121 11111
char:97 00
char:114 011
char:101 10
char:98 11101
char:97 00
char:119 01000
char:101 10
char:114 011
char:101 10
char:101 10
char:102 1100
char:39 11010110
char:107 11011
char:101 10
char:101 10
char:97 00
char:97 00
char:99 11100
char:101 10
char:101 10
char:102 1100
char:101 10
char:102 1100
char:101 10
char:114 011
char:101 10
char:101 10
char:97 00

char:114 011
char:101 10
char:98 11101
char:101 10
char:49 0100111101
char:56 010011011
char:102 1100
char:101 10
char:101 10
char:40 0100111010
char:53 010011010
char:53 010011010
char:41 0100111011
char:102 1100
char:114 011
char:101 10
char:97 00
char:80 1101010101
char:101 10
char:114 011
char:99 11100
char:103 0101
char:118 010010
char:101 10
char:101 10
char:101 10
char:97 00
char:102 1100
char:101 10
char:102 1100
char:97 00
char:101 10
char:101 10
char:114 011
char:102 1100
char:101 10
char:114 011

char:101 10
char:101 10
char:97 00
char:101 10
char:101 10
char:97 00
char:119 01000
char:114 011
char:114 011
char:121 11111
char:101 10
char:121 11111
char:97 00
char:114 011
char:101 10
char:84 11010111
char:114 011
char:97 00
char:101 10
char:101 10
char:97 00
char:114 011
char:98 11101
char:121 11111
char:99 11100
char:97 00
char:102 1100
char:101 10
char:97 00
char:107 11011
char:101 10
char:121 11111
char:97 00
char:107 11011
char:101 10
char:102 1100
char:114 011

char:97 00
char:97 00
char:114 011
char:107 11011
char:97 00
char:119 01000
char:114 011
char:114 011
char:101 10
char:97 00
char:98 11101
char:101 10
char:97 00
char:99 11100
char:114 011
char:101 10
char:112 11110
char:102 1100
char:103 0101
char:97 00
char:102 1100
char:101 10
char:97 00
char:101 10
char:101 10
char:112 11110
char:97 00
char:97 00
char:106 1101010111
char:114 011
char:101 10
char:97 00
char:97 00
char:98 11101
char:114 011
char:103 0101
char:101 10

char:119 01000
char:112 11110
char:97 00
char:112 11110
char:101 10
char:114 011
char:97 00
char:99 11100
char:102 1100
char:102 1100
char:101 10
char:101 10
char:87 1101010110
char:101 10
char:101 10
char:97 00
char:119 01000
char:97 00
char:107 11011
char:101 10
char:101 10
char:121 11111
char:112 11110
char:114 011
char:101 10
char:102 1100
char:103 0101
char:101 10
char:101 10
char:114 011
char:99 11100
char:101 10
char:97 00
char:103 0101
char:97 00
char:85 01001100
char:112 11110

char:114 011
char:101 10
char:114 011
char:101 10
char:101 10
char:112 11110
char:83 110100
char:97 00
char:97 00
char:103 0101
char:114 011
char:101 10
char:97 00
char:102 1100
char:121 11111
char:102 1100
char:97 00
char:102 1100
char:114 011
char:99 11100
char:97 00
char:98 11101
char:101 10
char:97 00
char:99 11100