Visualization:

All code is from geeksforgeeks.org

1. Easy code: <https://www.geeksforgeeks.org/capitalize-1st-character-of-all-words-having-at-least-k-characters/>

Prompt:

public static String capitalizeStr(String str, int K)

{

// Stores location of

// the 1st character of

// the current word

int ptr = 0;

char[] ch = str.toCharArray();

// Loop to traverse string

for (int i = 0; i < ch.length; i++) {

// If the current word

// ends at index i

if (ch[i] == ' ') {

// Update ptr

ptr = i + 1;

}

// Count of characters

// is at least K

else if (i - ptr + 1 >= K) {

ch[ptr] = Character.toUpperCase(ch[ptr]);

}

}

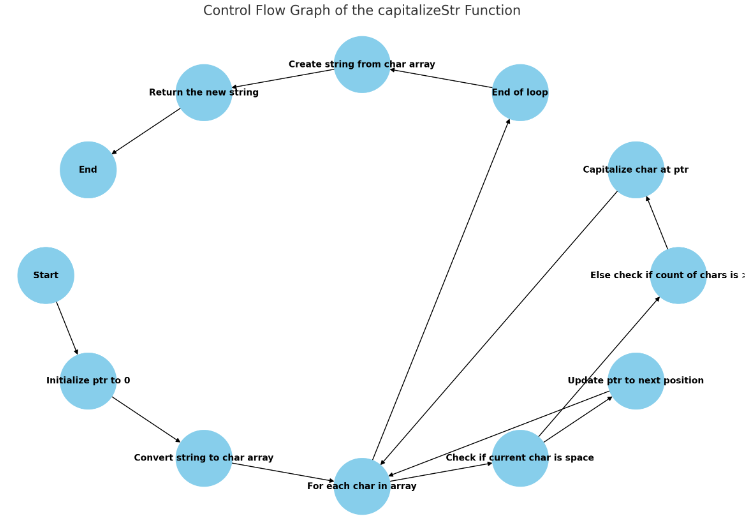
String s = new String(ch);

// Return answer

return s;

} Could you generate a control flow graph of the code in matplotlib format, ensuring that the nodes and edges do not overlap? On each node of the control flow graph, please use a short explanation sentence rather than raw code.

GPT output:



2. Easy code: <https://www.geeksforgeeks.org/implementation-of-bit-stuffing-and-bit-destuffing/>

Not good because GPT failed to generate a sentence

static void bitStuffing(int N, int arr[])

{

// Stores the stuffed array

int[] brr = new int[30];

// Variables to traverse arrays

int i, j, k;

i = 0;

j = 0;

// Loop to traverse in the range [0, N)

while (i < N) {

// If the current bit is a set bit

if (arr[i] == 1) {

// Stores the count of consecutive ones

int count = 1;

// Insert into array brr[]

brr[j] = arr[i];

// Loop to check for

// next 5 bits

for (k = i + 1; k < N && arr[k] == 1

&& count < 5;

k++) {

j++;

brr[j] = arr[k];

count++;

// If 5 consecutive set bits

// are found insert a 0 bit

if (count == 5) {

j++;

brr[j] = 0;

}

i = k;

}

}

// Otherwise insert arr[i] into

// the array brr[]

else {

brr[j] = arr[i];

}

i++;

j++;

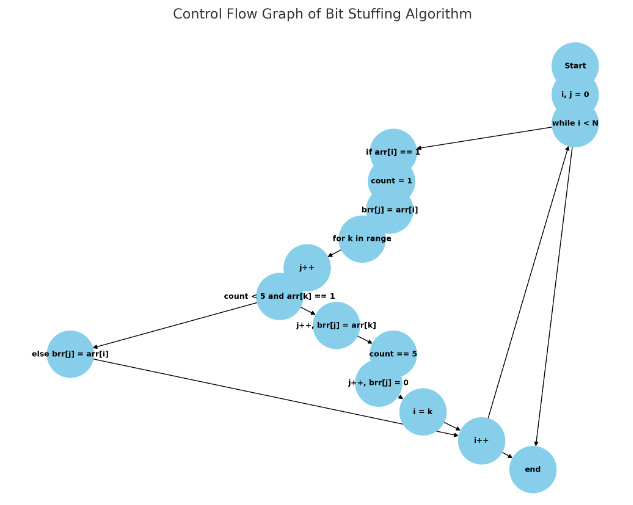
}

// Print Answer

for (i = 0; i < j; i++)

System.out.printf("%d", brr[i]);

} Could you generate a control flow graph of the code in matplotlib format, ensuring that the nodes and edges do not overlap? On each node of the control flow graph, please use a short explanation sentence rather than raw code.



3. Easy Code: <https://www.geeksforgeeks.org/java-program-for-converting-roman-numerals-to-decimal-lying-between-1-to-3999/>

Correct

Prompt: // Finds decimal value of a

// given roman numeral

int romanToDecimal(String str)

{

// Initialize result

int res = 0;

for (int i = 0; i < str.length(); i++)

{

// Getting value of symbol s[i]

int s1 = value(str.charAt(i));

// Getting value of symbol s[i+1]

if (i + 1 < str.length())

{

int s2 = value(str.charAt(i + 1));

// Comparing both values

if (s1 >= s2)

{

// Value of current symbol

// is greater or equalto

// the next symbol

res = res + s1;

}

else

{

// Value of current symbol is

// less than the next symbol

res = res + s2 - s1;

i++;

}

}

else

{

res = res + s1;

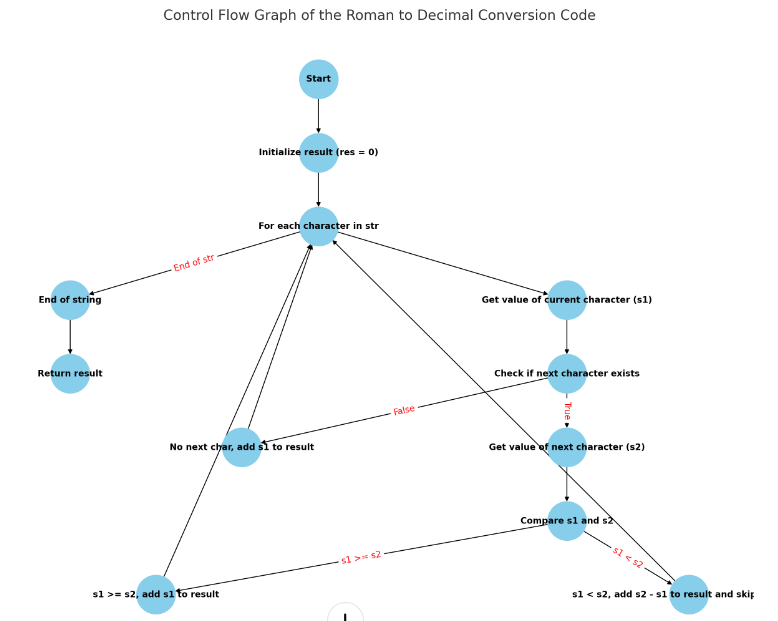
}

}

return res;

}

Could you generate a control flow graph of the code in matplotlib format, ensuring that the nodes and edges do not overlap? On each node of the control flow graph, please use a short explanation sentence rather than raw code.



4. Medium code

<https://www.geeksforgeeks.org/find-minimum-area-of-rectangle-formed-from-given-shuffled-coordinates/>

Correct

static int matchGame(long N)

{

// res variable stores the number of matchsticks

// initially picked by A

int res = (int)(N % 5);

// If res or N%5 is 0 then there is no chance of A

// to win the game

if (res == 0) {

return -1;

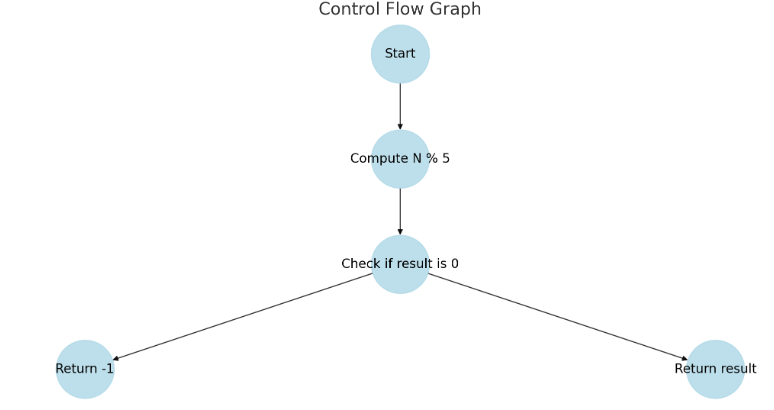
}

// else return the remainder value after dividing N

// by 5

return res;

} Could you generate a control flow graph of the code in matplotlib format, ensuring that the nodes and edges do not overlap? On each node of the control flow graph, please use a short explanation sentence rather than raw code.



5. Medium Code: <https://www.geeksforgeeks.org/bingo-sort-algorithm/>

Correct

Prompt:

// Function to sort the array

static int[] bingoSort(int[] vec, int n)

{

bingo = vec[0];

nextBingo = vec[0];

maxMin(vec, n);

int largestEle = nextBingo;

int nextElePos = 0;

while (bingo < nextBingo) {

// Will keep the track of the element position

// to

// shifted to their correct position

int startPos = nextElePos;

for (int i = startPos; i < n; i++) {

if (vec[i] == bingo) {

int temp = vec[i];

vec[i] = vec[nextElePos];

vec[nextElePos] = temp;

nextElePos = nextElePos + 1;

}

// Here we are finding the next Bingo

// Element for the next pass

else if (vec[i] < nextBingo)

nextBingo = vec[i];

}

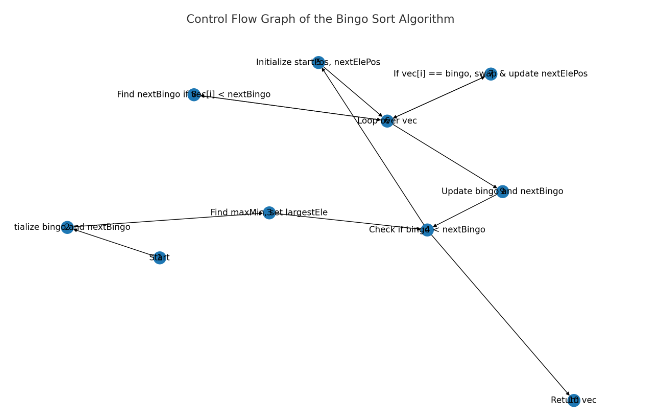
bingo = nextBingo;

nextBingo = largestEle;

}

return vec;

} Could you generate a control flow graph of the code in matplotlib format, ensuring that the nodes and edges do not overlap? On each node of the control flow graph, please use a short explanation sentence rather than raw code.



6. Medium code: <https://www.geeksforgeeks.org/day-stout-warren-algorithm-to-balance-given-binary-search-tree/>

Wrong: Move tmp to tmp.right

Prompt:

static int bstToVine(TreeNode grand)

{

int count = 0;

// Make tmp pointer to traverse

// and right flatten the given BST.

TreeNode tmp = grand.right;

// Traverse until tmp becomes NULL

while (tmp != null) {

// If left exist for node

// pointed by tmp then

// right rotate it.

if (tmp.left != null) {

TreeNode oldTmp = tmp;

tmp = tmp.left;

oldTmp.left = tmp.right;

tmp.right = oldTmp;

grand.right = tmp;

}

// If left dont exists

// add 1 to count and

// traverse further right to

// flatten remaining BST.

else {

count++;

grand = tmp;

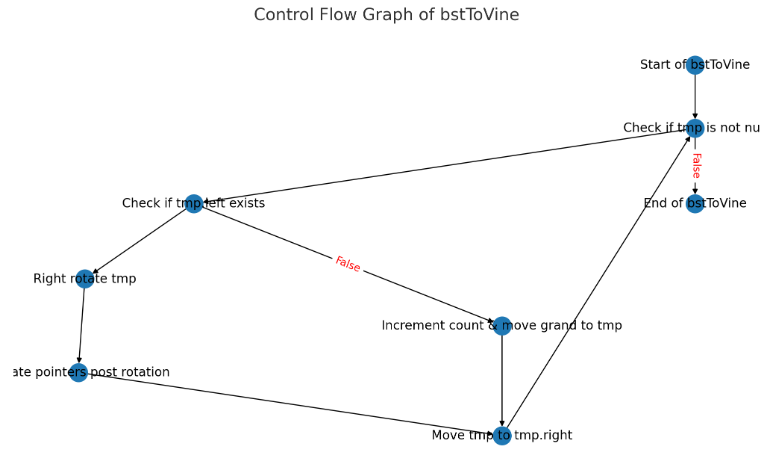
tmp = tmp.right;

}

}

return count;

}Could you generate a control flow graph of the code in matplotlib format, ensuring that the nodes and edges do not overlap? On each node of the control flow graph, please use a short explanation sentence rather than raw code.



7. Hard code:

<https://www.geeksforgeeks.org/find-minimum-area-of-rectangle-formed-from-given-shuffled-coordinates/>

Correct

public static int minArea(int[] A, int N)

{

// Sorting given array

Arrays.sort(A);

// Initializing answer to infinity

int ans = Integer.MAX\_VALUE;

// Initializing window from

// 0 to (N / 2) - 1

int low = 0, high = N / 2 - 1;

while (true) {

if (low == 0) {

// Updating answer for

// first slide

ans = (A[N / 2 - 1] - A[0])

\* (A[N - 1] - A[N / 2]);

}

// Break when slide reaches

// at end

else if (high == N - 1)

break;

else {

// Updating answer for

// i'th slide

ans = Math.min(ans, (A[high] - A[low])

\* (A[N - 1] - A[0]));

}

// Moving slide of size N / 2 by

// one position

low++;

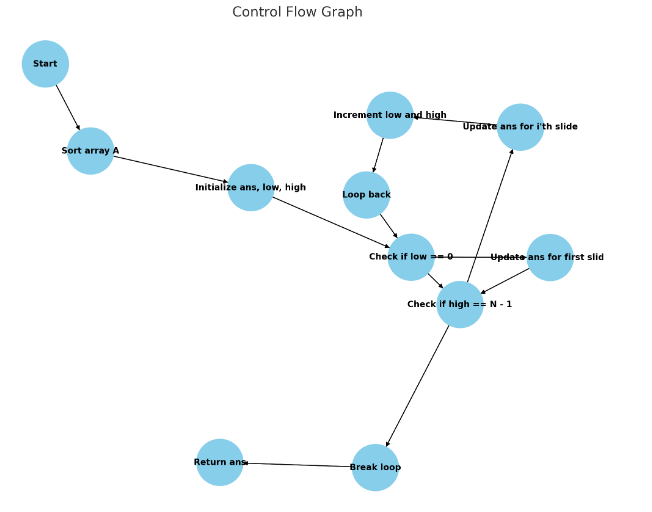
high++;

}

// Returning the answer

return ans;

}Could you generate a control flow graph of the code in matplotlib format, ensuring that the nodes and edges do not overlap? On each node of the control flow graph, please use a short explanation sentence rather than raw code.



8. Hard Code: <https://www.geeksforgeeks.org/inverse-ackermann-function/>

Correct

Prompt:

public static int inverseAckermann(int n)

{

// Check if the input is small enough

// to solve directly

if (n <= 4) {

return n;

}

// Divide the problem into

// two smaller problems

int a = inverseAckermann(n - 1);

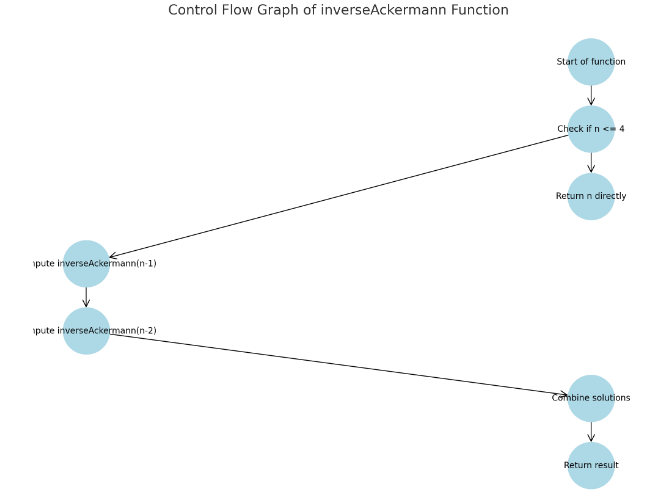
int b = inverseAckermann(n - 2);

// Combine the solutions of the

// two smaller problems

return a + b;

} Could you generate a control flow graph of the code in matplotlib format, ensuring that the nodes and edges do not overlap? On each node of the control flow graph, please use a short explanation sentence rather than raw code.



9. Hard Code: <https://www.geeksforgeeks.org/maximum-length-of-sequence-formed-from-cost-n/>

correct

Prompt:

// Function to find Maximum length of Sequence that can

// be formed from cost N

static void findMaximumLength(int N)

{

int low = 1, high = 1e9;

while (high - low > 1) {

int mid = low + (high - low) / 2;

// Check if cost for number of digits from 1 to

// N is less than equal to N

if (totalDigits(mid) <= N) {

// atleast mid will be the answer

low = mid;

}

else {

// ignore right search space

high = mid - 1;

}

}

// check if high can be the answer

if (totalDigits(high) <= N) {

System.out.println(high);

}

// else low can be the answer

else if (totalDigits(low) <= N) {

System.out.println(low);

}

// else answer will be zero.

else {

System.out.println(0);

}

}

Could you generate a control flow graph of the code in matplotlib format, ensuring that the nodes and edges do not overlap? On each node of the control flow graph, please use a short explanation sentence rather than raw code.

