

```

1  /* C program by Dave Russillo. Made on for CS1311. */
2  #include <stdio.h>
3  #include <stdlib.h>
4  #include <stdbool.h>
5
6
7  struct Node {
8      int value;
9      struct Node *left;
10     struct Node *right;
11 };
12
13 void init_tree(struct Node *root) {
14     void init_tree_helper(struct Node *current, int current_depth) {
15         // initialize child nodes
16         current->left = malloc(sizeof(struct Node));
17         current->right = malloc(sizeof(struct Node));
18         // keep initializing nodes until depth 3;
19         // iterate only until depth 2 because children are initialized
20         if(current_depth < 2) {
21             init_tree_helper(current->left, current_depth + 1);
22             init_tree_helper(current->right, current_depth + 1);
23         }
24     }
25     init_tree_helper(root, 0);
26
27     // add second to last row (depth 4)
28     root->left->left->left->left = malloc(sizeof(struct Node));
29     root->left->right->right->right = malloc(sizeof(struct Node));
30     root->right->left->left->left = malloc(sizeof(struct Node));
31     root->right->right->right->right = malloc(sizeof(struct Node));

```

```

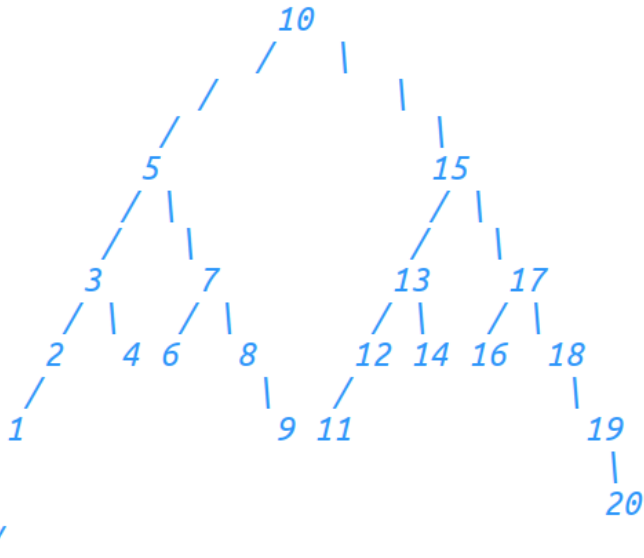
32 // add last row (depth 5)
33 root->right->right->right->right->right = malloc(sizeof(struct Node));
34 /*
35  * Result:
36  *
37  *
38  *
39  *
40  *
41  *
42  *
43  *
44  *
45  *
46  *
47  *
48  *
49  */
50 */
51 }
52
53
54 void populate_tree(struct Node *root) {
55     void populate_tree_helper(struct Node *current, int difference, int set_value) {
56         current->value = set_value;
57         // set next difference to half of current
58         difference = difference / 2;
59         // minimum difference must be 1
60         if(difference == 0) {
61             difference = 1;
62         }

```

```

63 if(current->left != NULL) {
64     // subtract difference on left
65     populate_tree_helper(current->left, difference, set_value - difference);
66 }
67 if(current->right != NULL) {
68     // add difference on right
69     populate_tree_helper(current->right, difference, set_value + difference);
70 }
71 }
72 populate_tree_helper(root, 10, 10);
73 /*
74  * Result:
75  *
76  *
77  *
78  *
79  *
80  *
81  *
82  *
83  *
84  *
85  *
86  *
87  *
88  *
89  */
90 }
91
92
93 void print_preorder(struct Node *current) {

```



```
94 // print current value
95 printf("%d ", current->value);
96
97 // iterate on left node
98 if(current->left != NULL) {
99     print_preorder(current->left);
100 }
101
102 // iterate on right node
103 if(current->right!= NULL) {
104     print_preorder(current->right);
105 }
106 }
107
108
109 void print_inorder(struct Node *current) {
110     // iterate on left node
111     if(current->left != NULL) {
112         print_inorder(current->left);
113     }
114
115     // print current value
116     printf("%d ", current->value);
117
118     // iterate on right node
119     if(current->right!= NULL) {
120         print_inorder(current->right);
121     }
122 }
123
124
```

```

125 void print_postorder(struct Node *current) {
126     // iterate on left node
127     if(current->left != NULL) {
128         print_postorder(current->left);
129     }
130
131     // iterate on right node
132     if(current->right != NULL) {
133         print_postorder(current->right);
134     }
135
136     // print current value
137     printf("%d ", current->value);
138 }
139
140
141 void take_user_input(struct Node *current) {
142     char choice = ' ';
143     bool valid_choice = false;
144
145     printf("
146     " | ----- | \n"
147     " | I think your number is %2d | \n"
148     " | ----- | \n"
149     " | \n"
150     " | Am I right? | \n"
151     " | ===== | \n"
152     " | a) Yes | \n"
153     " | b) My number is higher | \n"
154     " | c) My number is lower | \n"
155     " | ----- | \n\n", current->value);

```

```

156 while(!valid_choice) {
157     printf("..? ");
158     choice = getchar();
159     valid_choice = choice == 'a' || choice == 'b' || choice == 'c';
160     while(getchar() != '\n');
161     if(!valid_choice) {
162         printf("Invalid answer! Try again.\n");
163     }
164 }
165 if(choice == 'c') {
166     if(current->left != NULL) {
167         take_user_input(current->left);
168     } else {
169         printf("Invalid number!\n");
170     }
171 } else if(choice == 'b') {
172     if(current->right != NULL) {
173         take_user_input(current->right);
174     } else {
175         printf("Invalid number!\n");
176     }
177 } else {
178     printf("I guessed right! Your number is %d.\n\n", current->value);
179 }
180 }
181
182
183 int main(void) {
184     struct Node *root = malloc(sizeof(struct Node));
185     init_tree(root);
186     populate_tree(root);
187
188     print_preorder(root);
189     printf("\n");
190     print_inorder(root);
191     printf("\n");
192     print_postorder(root);
193     printf("\n\n");
194
195     printf("Think of a number between 1 and 20 and I will try to guess it.\n"
196           "Press enter to start ...");
197     while(getchar() != '\n');
198     printf("\n\n");
199
200     take_user_input(root);
201
202     return 0;
203 }
204

```

```
10 5 3 2 1 4 7 6 8 9 15 13 12 11 14 17 16 18 19 20
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
1 2 4 3 6 9 8 7 5 11 12 14 13 16 20 19 18 17 15 10
```

Think of a number between 1 and 20 and I will try to guess it.
Press enter to start ...

I think your number is 10

Am I right?

- =====
- a) Yes
 - b) My number is higher
 - c) My number is lower

..? c

I think your number is 5

Am I right?

- =====
- a) Yes
 - b) My number is higher
 - c) My number is lower

..? b

I think your number is 7

Am I right?

- =====
- a) Yes
 - b) My number is higher
 - c) My number is lower

..? c

I think your number is 6

Am I right?

- =====
- a) Yes
 - b) My number is higher
 - c) My number is lower

..? a

I guessed right! Your number is 6.

I think your number is 10

Am I right?

- =====
- a) Yes
 - b) My number is higher
 - c) My number is lower

..? b

I think your number is 15

Am I right?

- =====
- a) Yes
 - b) My number is higher
 - c) My number is lower

..? b

I think your number is 17

Am I right?

- =====
- a) Yes
 - b) My number is higher
 - c) My number is lower

..? b

I think your number is 18

Am I right?

- =====
- a) Yes
 - b) My number is higher
 - c) My number is lower

..? b

I think your number is 19

Am I right?

- =====
- a) Yes
 - b) My number is higher
 - c) My number is lower

..? b

I think your number is 20

Am I right?

- =====
- a) Yes
 - b) My number is higher
 - c) My number is lower

..? b

Invalid number!