**CPSC 6109:** [**Advanced**](https://colstate.view.usg.edu/d2l/lp/ouHome/home.d2l?ou=1218642) **Algorithms**

**Spring 2018**

**Assignment #4**

**Student: Lu Lin**

**Chapter 16. Greedy Algorithms**

**Due: 11:59 PM Tuesday March 6**

Do the following exercises/problems. Each problem is worth 50 points with a total of 100 points.

1. Consider the activity-selection problem in Chapter 16 (Section 16.1) for scheduling several competing activities that require exclusive use of a common resource, with a goal of selecting a maximum-size set of mutually compatible activities. Suppose we have a set ***S*** = {a1, a2, …, an} of ***n*** proposed ***activities*** that wish to use a common resource. The resource can serve only one activity at a time. Assume that each activity is represented by an interval. For example, the activity aj = [sj, fj) starts at sj and finishes at fj. The length of the activity aj = [sj, fj) is defined as fj - sj.

Sort all the activities in the increasing order of their interval lengths. Suppose that instead of always selecting the first activity to finish as in the textbook, we instead select the activity with the shortest interval that is compatible with all previously selected activities.

1. Give a counter example to show that this greedy approach DOES NOT always yield an optimal solution.
2. What is the solution given by this greedy approach on the counter example?
3. What is the optimal solution of this counter example?

Solutions:

1. This counter example is as following. As an approach of selecting the activity of shortest interval from those that are compatible with all previously selected activates.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| i | 1 | 2 | 3 | 4 | 5 |
| si | 0 | 2 | 3 | 4 | 6 |
| fi | 3 | 4 | 6 | 6 | 9 |
| interval | 3 | 2 | 3 | 2 | 3 |

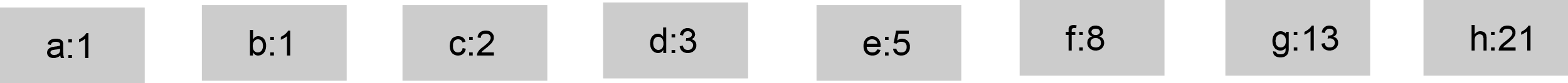
1. This approach selects {a2, a4} as solution.
2. The optimal solution selects {a1, a3, a5} as solution.
3. Problem **16.3-3** on page 436. What is an optimal Huffman code for the following set of frequencies, based on the first 8 Fibonacci numbers?

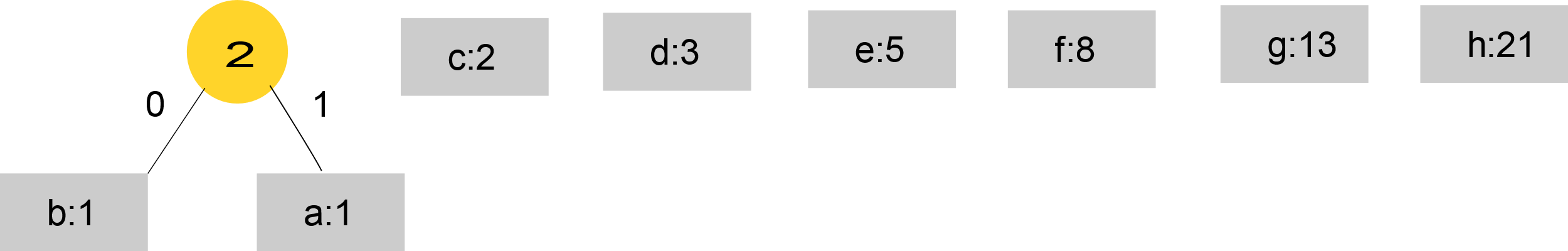
a: 1 b:1 c:2 d:3 e:5 f:8 g:13 h:21

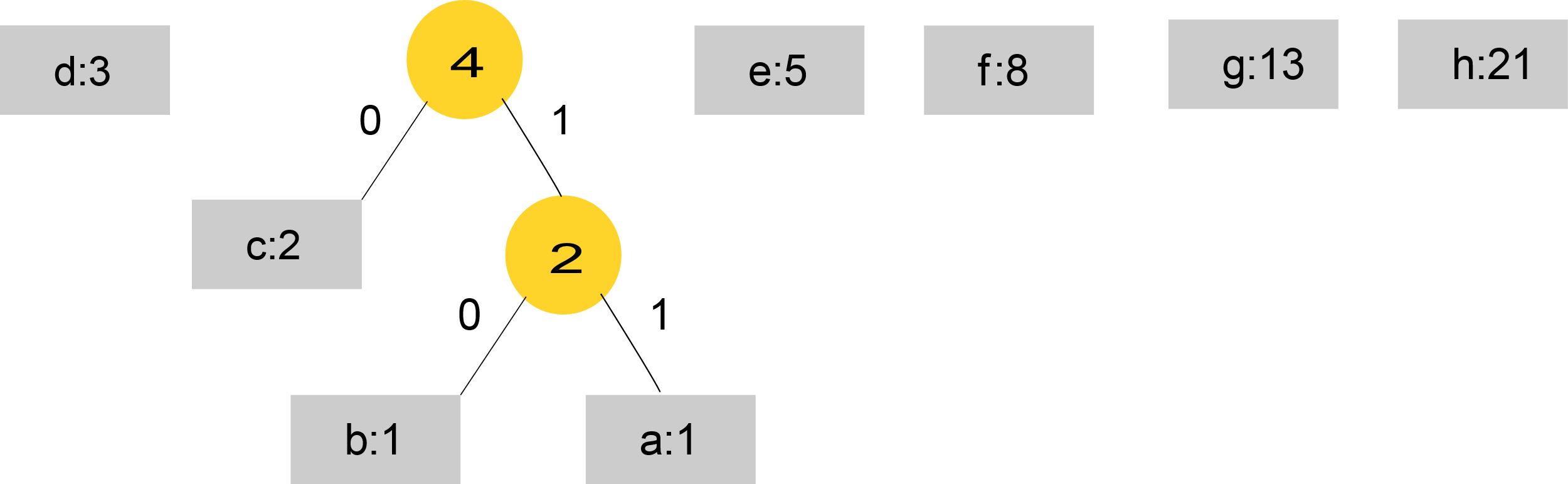
Can you generalize your answer to find the optimal code when the frequencies are the first n Fibonacci numbers?

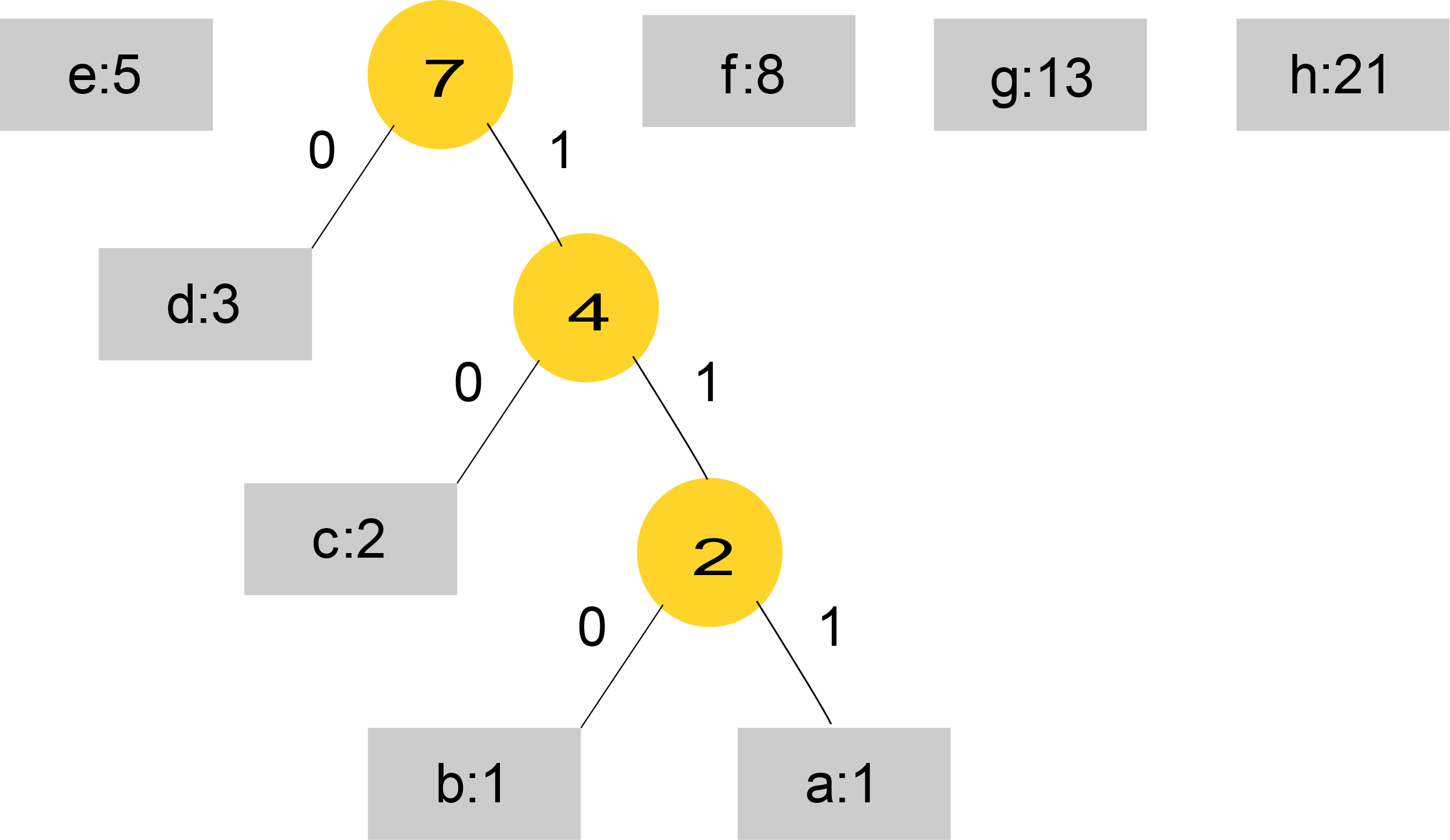
Solution:

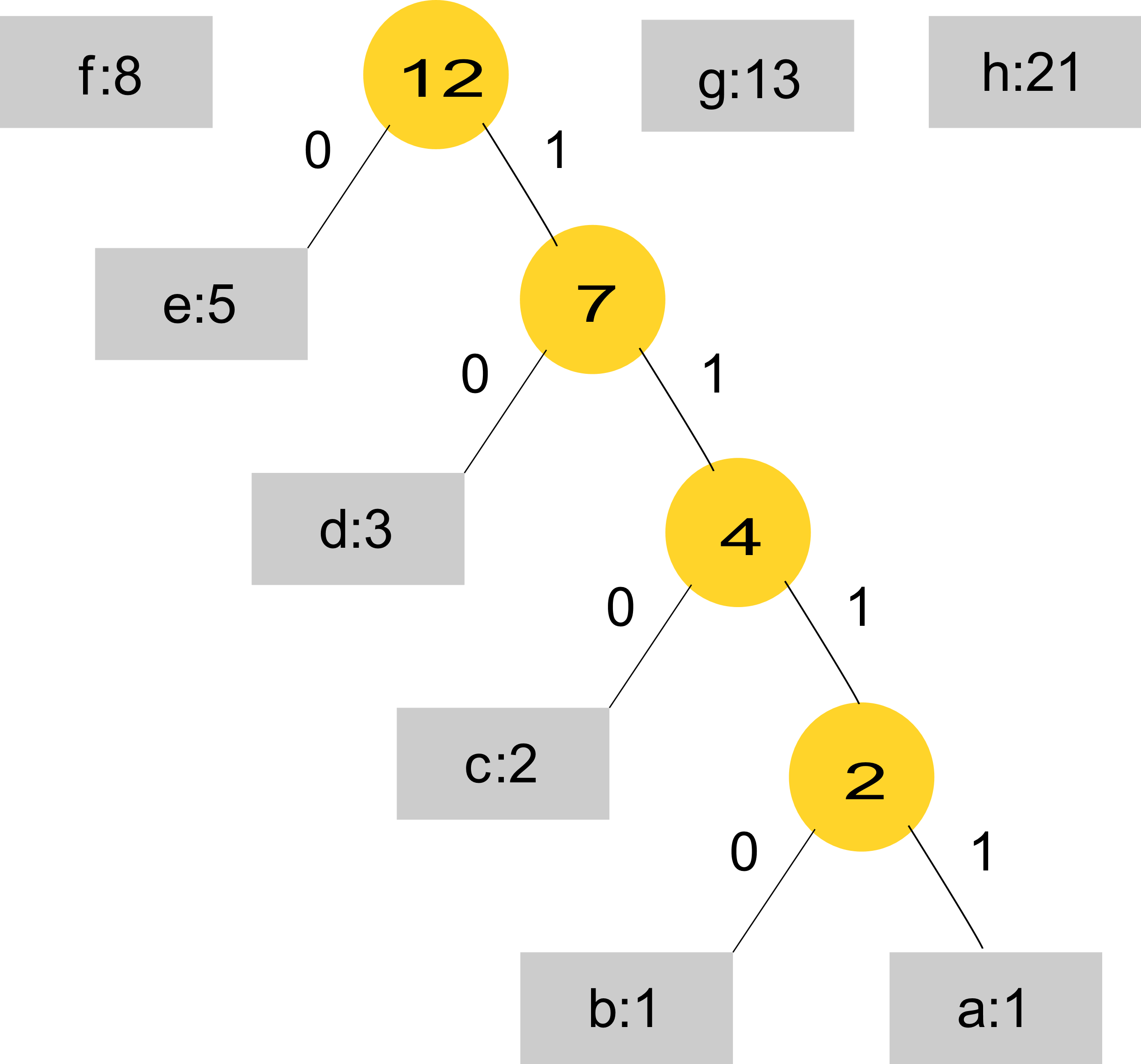
For the given Fibonacci numbers n = 8, steps for constructing a Huffman tree are as follows:

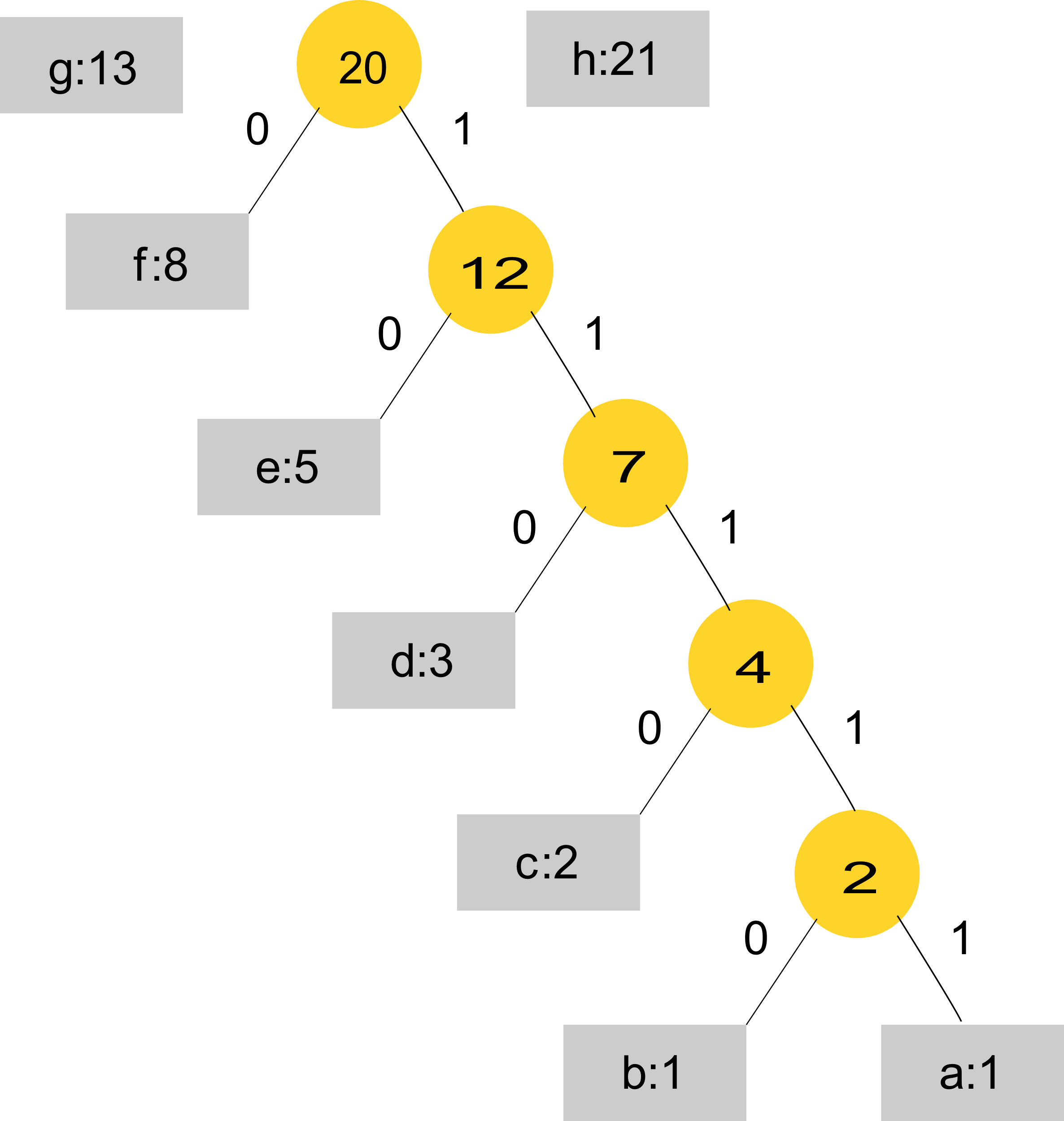


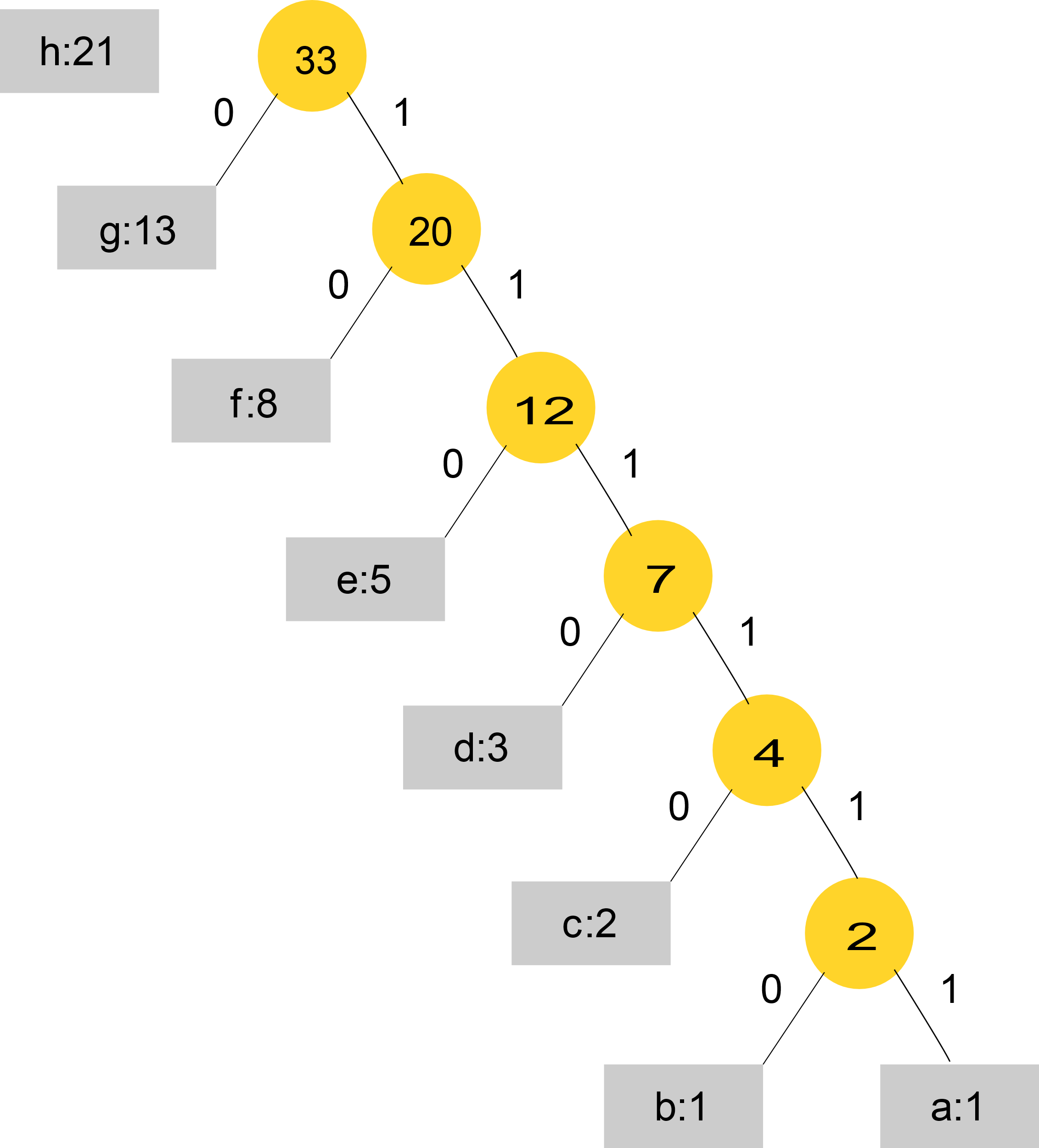


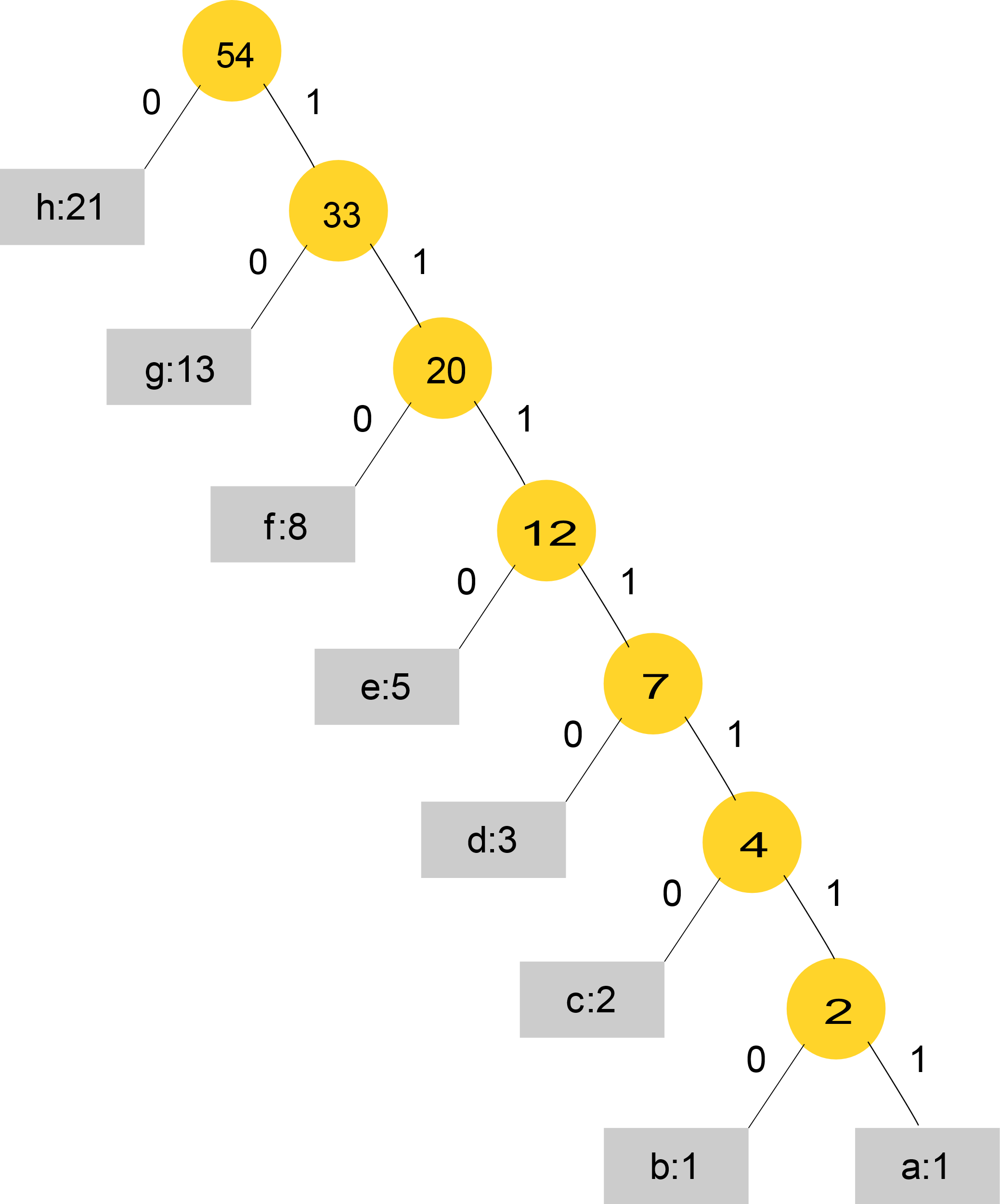












The Huffman codes for 8 Fibonacci numbers are as following:

h:0

g:10

f:110

e:1110

d:11110

c:111110

b:1111110

a:1111111

In general, the first n Fibonacci numbers have optimal Huffman codes as:

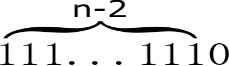
Fn:0

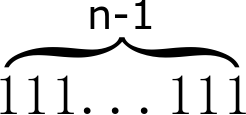
Fn-1:10

Fn-2:110

Fn-3:1110

F3: 

F2:

F1: 

|  |  |
| --- | --- |
| Submission Feedback |  |
| 1) wrong counterexample -10 2) | |

**CPSC 6109:** [**Advanced**](https://colstate.view.usg.edu/d2l/lp/ouHome/home.d2l?ou=1218642) **Algorithms**

**Spring 2018**

**Assignment #4**

**Solutions**

1. Consider the activity-selection problem in Chapter 16 (Section 16.1) for scheduling several competing activities that require exclusive use of a common resource, with a goal of selecting a maximum-size set of mutually compatible activities. Suppose we have a set ***S*** = {a1, a2, …, an} of ***n*** proposed ***activities*** that wish to use a common resource. The resource can serve only one activity at a time. Assume that each activity is represented by an interval. For example, the activity aj = [sj, fj) starts at sj and finishes at fj. The length of the activity aj = [sj, fj) is defined as fj - sj.

Sort all the activities in the increasing order of their interval lengths. Suppose that instead of always selecting the first activity to finish as in the textbook, we instead select the activity with the shortest interval that is compatible with all previously selected activities.

1. Give a counter example to show that this greedy approach DOES NOT always yield an optimal solution.   
   **S={a1, a2, a3},**

**a1=[0, 3), a2=[2, 4), a3=[3, 7).**

1. What is the solution given by this greedy approach on the counter example?  
   **{a2}**
2. What is the optimal solution of this counter example?   
   **{a1, a3}**

2. Problem **16.3-3** on page 436.

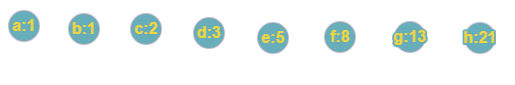


Figure 1 Initial



Figure 2 Nodes a and b combined

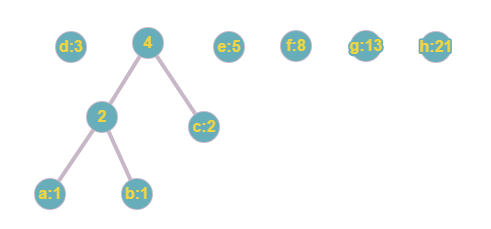


Figure 3 Node 2 and c combined

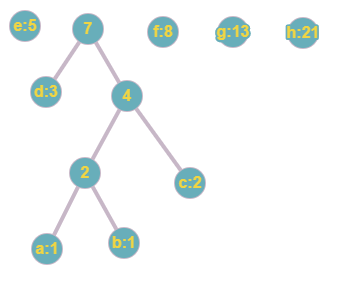


Figure 4 Node d and 4 combined

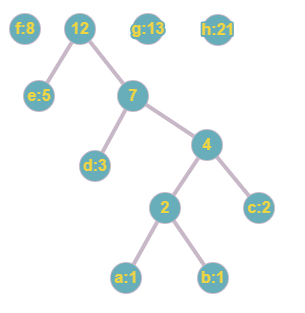


Figure 5 Node e and 7 combined

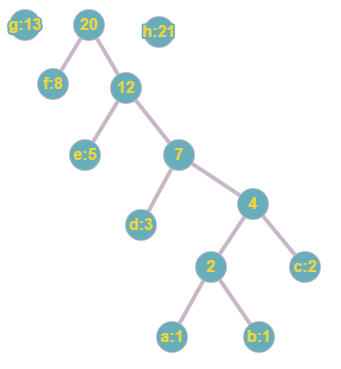


Figure 6 Node 8 and 12 combined

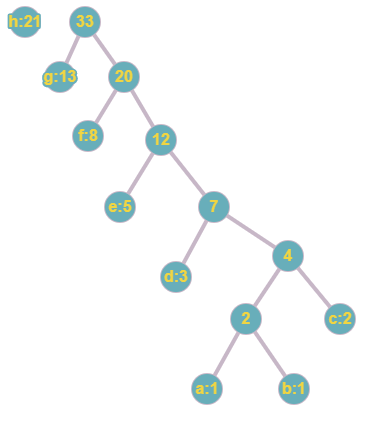


Figure 7 Node g and 20 combined

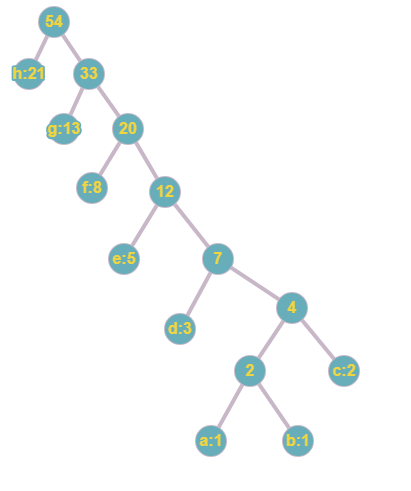


Figure 8 Huffman tree of first 8 Fibonacci numbers

Notice that after combining a and b, at each step, we combine the next Fibonacci number with the combined node from its previous step.

We will argue that this is true for first Fibonacci n numbers for any n ≥ 3.

**To do so, it suffices to show that**

.

Clearly, when n ≥ 3.

Next, we prove by way of induction that for n ≥ 3, that is .

When n = 3, we see that .

When n = 4, .

Suppose for n > 4, for any k such that 4 ≤ k < n.

Then we have

, which is what we needed to prove.

Since we know that the combined node at any given step is one of the two minimum frequency nodes to be combined at the next step, it is easy enough to argue that for arbitrary n, a Huffman tree for first n Fibonacci numbers will look like this (or with node 2 and f3 in swapped position):

