Reset deadlines

How it works

You've successfully completed the course! To access up-to-date course material, please reset your deadlines.

Reset my deadlines

Discussion Forums

Week 4

SUBFORUMS

All

Assignment: Programming Assignment 4

← Week 4

Advanced Problem: Reschedule the Exams

Alexander S. Kulikov Instructor · 4 years ago

Please use this thread to discuss Reschedule the Exams advanced problem (make sure to review forum rules before posting).

0 Upvotes Reply Follow this discussion

Earliest **Top** **Most Recent**

LW **LI Wen** · 4 years ago

TIPS: consider about 2-SAT problem,here are 3 constraints

1. each node can **not** be the same color of its initial state.
2. each node should be **exactly one** of another two colors differ from its original.
3. each node can **not** be of the same color of its adjacent node.

3 Upvotes Reply

Seungil Lee · 4 years ago

I passed test case #3, but got time limit at test case #5. I implemented backtracking algorithm. Did I choose wrong algorithm?

0 Upvotes Hide 3 Replies

Vivekanand Ganapathy Nagarajan · 4 years ago

Yes, you need to choose a different algorithm .

0 Upvotes

Seungil Lee · 4 years ago

I now have an idea. Is this problem reduced to an O(n) algorithm?

0 Upvotes

Seungil Lee · 4 years ago

I passed it. Thanks for the tips.

0 Upvotes

Reply

Reply

Seungil Lee · 4 years ago

It seems that others think this is easiest assignment, but I'm stuck at test case #3, getting "Wrong Answer". Could you help me to post some test cases or any hint how you did stress test?

1 Upvote Reply

Sam Handelman · 4 years ago

Ironically, I found this problem to be the easiest of them all... For anyone having trouble (and have successfully completed the other problems already), don't overthink it. Think about what two major sets of constraints need to be set, based on the problem description, how we can easily express this, and subsequently compute a satisfying solution if there is one.

0 Upvotes Hide 13 Replies

Kota Mori · 4 years ago

I agree. It is easy once you notice the appropriate reduction. To me, a line the problem set was critical clue: "You need to reduce this problem to another problem you already know how to solve."

0 Upvotes

Vivekanand Ganapathy Nagarajan · 4 years ago

@Kota, Sam - Did you solve this using backtracking .

I am wondering because my solution fails with StackOverFlow exception because of the depth of recursion .

This is quite possible because each of the tests generates a lot of clauses.

The sample tests generates 31 clauses to be precise.

0 Upvotes

Kota Mori · 4 years ago · Edited

Hi, I am on C++ and have never experienced stack overflow exception since I started to write everything by vectors instead of arrays. I don't know why or if this is good or bad.

My answer uses a lot of recursive procedure and works well. As the matter of fact, I like recursion for some reason.

I just copied my own answer from a previous question modified the input and output handling appropriately. Since the grader of this course is well designed, if your code worked in the previous question, then your algorithm is very likely to be okay.

So, If it fails, I would suspect the input/output part in the first place.

0 Upvotes

Vivekanand Ganapathy Nagarajan · 4 years ago

My solution now fails due to timing issues in TC #5.

After performing some benchmarking , i found out the bottle neck in my solution is in the recursive method while updating clauses when making variable assignments.

In every update, I construct new clauses and go through every clause to check if the variable belongs to it . I see no way around this because you need a handle to the original clauses object in case you return from back tracking step and hence i do not modify the original clauses.

0 Upvotes

Kota Mori · 4 years ago · Edited

Can you describe your approach? It seems yours differs from mine because I did not have update clauses.

0 Upvotes

Vivekanand Ganapathy Nagarajan · 4 years ago

Each time I make an assignment to a variable , I iterate through the list of clauses to check if the variable belongs to the clause . E.g) If I make as assignment x=0, I iterate through each clause to check if variable is contained in clause , copying over non matching variables to the new copy of the clause .

0 Upvotes

Kota Mori · 4 years ago

Hmm, sounds very complex. I used an algorithm learned in one of the previous problems. You don't need to update clauses at all.

0 Upvotes

Kota Mori · 4 years ago

That said, can you explain what kind of clauses you constructed for the problem, and how you tried to find a solution to them?

0 Upvotes

Vivekanand Ganapathy Nagarajan · 4 years ago

Hi Kota,

I constructed the clauses using the same solution from Week 3 on VertexColoring.

I then used the recursive backtracking algorithm as mentioned in the lecture notes.

Eg)

SolveSAT(F)

if F has no clauses:

```
return “sat”

if F contains an empty clause:

return “unsat”

x ← unassigned variable of F

if SolveSAT(F [x ← 0]) = “sat”:

return “sat”
```

If you can see above, in the step, **SolveSAT(F [x ← 0]) = “sat”:**

that is where I update the clauses, iterating through the list of clauses to satisfy the assignment and hence create new clauses.

If that is not the case, how did you implement this step in the algorithm.

⇧ 0 Upvotes

Kota Mori · 4 years ago

I see! There is actually an easier solution. As you might have seen already, in this problem, there are two types of clauses.

First, if the current color of x is "R", then you need to change it to either G or B, so: (x=G or x=B) & (x!=G or x!=B).

Second, if x and y are adjacent, then we need to color them different, so: (x!=R or y!=R) & (x!=G or y!= G) & (x!=B or y!=B).

For this type of SAT problem we learned a more efficient algorithm.

⇧ 4 Upvotes

Vivekanand Ganapathy Nagarajan · 4 years ago

Thanks Kota for that insight . Using this additional information reduced the no of clauses required .

You mentioned we learned a more efficient algorithm, are you referring to the backtracking one or something else.

⇧ 0 Upvotes

Kota Mori · 4 years ago

Something else. Just review this week's pset and you will realize!

0 Upvotes

Seungil Lee · 4 years ago

@Kota Mori Is it Local Search algorithm?

0 Upvotes

Reply

Reply

YX **Yuan Xiao** · 4 years ago

I think in the Python starter solution,

```
colors = input().split()
```

should be changed to

```
colors = input()
```

2 Upvotes Reply

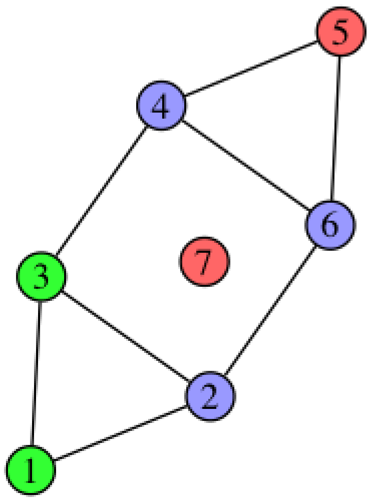
DT **Daniel Trebbien** · 4 years ago

If one of the students is not close friends with any other student, don't forget that they still need to be assigned a different color.

Here is a test case for this:

Input:

```
7 8
GBGBRBR
1 3
3 2
3 4
4 6
2 6
4 5
5 6
1 2
```



Output:

```
BGRGBRB
```

If the last letter of the output is 'R', then student 7 has not been recolored.

5 Upvotes Reply

1

Reply

Reply