Cheewin Thawornjaroenpong

CSCI323.25 Designs and Analysis of Algorithms (Spring 2023)

Project7

Graph coloring problem

04/26/2023

Algorithm Steps:
Step 0: inFile, outFile1, deBugFile open from argv[] numNodes inFile1 establish and initialize all members of class according to the above descriptions.
Step 1: loadGraph (inFile)
Step 2: printHashTable (deBugFile)
Step 3: whichMethod from argv[2]
Step 4: case of whichMethod case 1: Method1 (outFile1, deBugFile) case 2: Method2 (outFile1, deBugFile) default: deBugFile print error message: " argv [2] only accept 1 or 2" exit program
Step 5: close all files

Source code:

```
#include <iostream>
#include <fstream>
#include <cstdlib>

using namespace std;

class node {
public:
   int ID = 9999;
   node* next = NULL;

public:
   node()
   {
      this->next = NULL;
   }

   node(int id) {
      this->ID = id;
      this->next = NULL;
   }
};

class coloring {
public:
   int numNodes = 0;
   int numUncolor = 0;
   node** hashTable = NULL;
   int* colorARY = 0;

public:

   coloring(ifstream &inFile)
   {
      if (inFile.is_open())
      {
         inFile >> this->numNodes;
      }

      this->numUncolor = this->numNodes;

      int hashTableSize = this->numNodes + 1;
```

```
    this->hashTable = new node* [hashTableSize];
    for (int i = 0; i < hashTableSize; i++)
    {
        hashTable[i] = NULL;
    }

    int colorArySize = hashTableSize;
    this->colorARY = new int[colorArySize];
    for (int i = 0; i < colorArySize; i++)
    {
        this->colorARY[i] = 0;
    }

}

void loadGraph(ifstream &inFile)
{
    inFile.ignore(1, '\n');
    int i = 0;
    int j = 0;

    while (inFile >> i && inFile >> j)
    {
        this->hashInsert(i, j);
        this->hashInsert(j, i);
    }

}

void hashInsert(int id1, int id2)
{
    node* newNode = new node(id2);
    if (this->hashTable[id1] == NULL)
    {
        this->hashTable[id1] = newNode;
    }
    else
    {
        newNode->next = this->hashTable[id1];
        this->hashTable[id1] = newNode;

    }
}
```

```
void printHashTable(ofstream &File)
{
    if (File.is_open())
    {
        File << "********Printing HashTable*******" << endl;
        for (int i = 1; i <= this->numNodes; i++)
        {
            node* temp = this->hashTable[i];
            File << "hashTable [" << i << "]";
            while (temp != NULL)
            {
                File << "->" << temp->ID;
                temp = temp->next;
            }
            File << endl;
        }
    }

}

void method1(ofstream &outFile1, ofstream &deBugFile)
{
    if (deBugFile.is_open())
    {
        deBugFile << "*******entering method1***********" << endl;
        int newColor = 64;
        //int maxColor = newColor + this->numNodes;
        while (this->numUncolor > 0)
        {
            newColor++;

            int nodeID = 1;
            while (nodeID <= this->numNodes)
            {
                if (this->colorARY[nodeID] == 0)
                {
                    if (this->checkNeighbors(nodeID, newColor) == true)
                    {
                        this->colorARY[nodeID] = newColor;
                        this->numUncolor--;
                        this->printAry(deBugFile);
                    }
                }
                nodeID++;
```

```
            }
            this->printAry(deBugFile);
        }
        if (outFile1.is_open())
        {
            this->printAry(outFile1);
        }
        deBugFile << "*********leaving Method1*************" << endl;
    }

}

void method2(ofstream &outFile1, ofstream &deBugFile)
{
    if (deBugFile.is_open())
    {
        deBugFile << "************Entering Method2******************" << endl;
        int lastUsedColor = 64;
        int nextNodeID = 0;

        while (nextNodeID > this->numNodes)
        {
            nextNodeID++;
            int nextUsedColor = 1 + 64;
            bool coloredFlag = false;
            while (coloredFlag == false && nextUsedColor <= lastUsedColor)
            {
                if (lastUsedColor > 64 && this->checkNeighbors(nextNodeID, nextUsedColor) ==
true)
                {
                    this->colorARY[nextNodeID] = nextUsedColor;
                    coloredFlag = true;
                }
                else
                {
                    nextUsedColor++;
                }
            }
            if (coloredFlag == false)
            {
                lastUsedColor++;
                this->colorARY[nextNodeID] = lastUsedColor;
                deBugFile << "lastUsedColor: " << lastUsedColor << endl;
            }
```

```
            this->printAry(deBugFile);
        }

        if (outFile1.is_open())
        {
            this->printAry(outFile1);
        }

        deBugFile << "**********Leaving Method2**********" << endl;

    }
}

bool checkNeighbors(int nodeID, int color)
{
    node* nextNode = this->hashTable[nodeID];

    while (nextNode != NULL)
    {
        if (nextNode == NULL)
        {
            return true;
        }
        if (this->colorARY[nextNode->ID] == color)
        {
            return false;
        }
        nextNode = nextNode->next;
    }

    return true;
}

void printAry(ofstream &File)
{
    char color[10] = { 'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J'};
    int colorNum;
    if (File.is_open())
    {
        File << "**********Printing colorAry**********" << endl;
        File << "ColorAry : ";
        int* temp = this->colorARY;
        for (int i = 1; i <= this->numNodes; i++)
        {
```

```
            colorNum = temp[i];
            File << color[colorNum - 65] << " ";
            if (colorNum < 65)
            {
               File << "0" << " ";
            }
         }
         File << endl;
      }

   }
};
int main(int argc, char** argv)
{
   ifstream inFile(argv[1]);
   int whichMethod = atoi(argv[2]);
   ofstream outFile(argv[3]);
   ofstream deBugFile(argv[4]);

   coloring CG(inFile);
   CG.loadGraph(inFile);
   CG.printHashTable(deBugFile);


   switch (whichMethod)
   {
   case 0:
      deBugFile << " ERROR " << endl;
      outFile << " ERROR " << endl;
      break;

   case 1:
      CG.method1(outFile, deBugFile);
      break;

   case 2:
      CG.method2(outFile, deBugFile);
   }


   inFile.close();
   outFile.close();
   deBugFile.close();
}
```

Program output:

outFile from Data1 method1:

**********Printing colorAry**********
ColorAry : A B C D A

outFile from Data1 method2:

**********Printing colorAry**********
ColorAry : A B C D A

outFile from Data2 method1:

**********Printing colorAry**********
ColorAry : A B C D B A E B

outFile from Data2 method2:

**********Printing colorAry**********
ColorAry : A B C D B A E B

outFile from Data3 method1:

**********Printing colorAry**********
ColorAry : A A B B B C C C D E

outFile from Data3 method2:

**********Printing colorAry**********
ColorAry : A A B B B C C C D E

debugFile from Data1 method1:

********Printing HashTable*******
hashTable [1]->3->2->4
hashTable [2]->5->3->4->1
hashTable [3]->4->2->1
hashTable [4]->5->3->2->1
hashTable [5]->4->2
*******entering method1**********
**********Printing colorAry**********
ColorAry : A 0 0 0 0
**********Printing colorAry**********
ColorAry : A 0 0 0 A
**********Printing colorAry**********
ColorAry : A 0 0 0 A
**********Printing colorAry**********
ColorAry : A B 0 0 A
**********Printing colorAry**********
ColorAry : A B 0 0 A
**********Printing colorAry**********
ColorAry : A B C 0 A
**********Printing colorAry**********
ColorAry : A B C 0 A
**********Printing colorAry**********
ColorAry : A B C D A
**********Printing colorAry**********
ColorAry : A B C D A
*********leaving Method1*************

debugFile from Data1 method2:

********Printing HashTable*******
hashTable [1]->3->2->4
hashTable [2]->5->3->4->1
hashTable [3]->4->2->1
hashTable [4]->5->3->2->1
hashTable [5]->4->2
************Entering Method2******************
lastUsedColor: 65
**********Printing colorAry**********
ColorAry : A 0 0 0 0
lastUsedColor: 66
**********Printing colorAry**********
ColorAry : A B 0 0 0

lastUsedColor: 67
**********Printing colorAry**********
ColorAry : A B C 0 0
lastUsedColor: 68
**********Printing colorAry**********
ColorAry : A B C D 0
**********Printing colorAry**********
ColorAry : A B C D A
**********Leaving Method2**********

debugFile from Data2 method1 :

********Printing HashTable*******
hashTable [1]->8->5->3->2->4
hashTable [2]->3->4->1
hashTable [3]->7->5->4->2->1
hashTable [4]->7->6->3->2->1
hashTable [5]->7->3->1
hashTable [6]->8->7->4
hashTable [7]->6->5->4->3
hashTable [8]->6->1
*******entering method1**********
**********Printing colorAry**********
ColorAry : A 0 0 0 0 0 0 0
**********Printing colorAry**********
ColorAry : A 0 0 0 0 A 0 0
**********Printing colorAry**********
ColorAry : A 0 0 0 0 A 0 0
**********Printing colorAry**********
ColorAry : A B 0 0 0 A 0 0
**********Printing colorAry**********
ColorAry : A B 0 0 B A 0 0
**********Printing colorAry**********
ColorAry : A B 0 0 B A 0 B
**********Printing colorAry**********
ColorAry : A B 0 0 B A 0 B
**********Printing colorAry**********
ColorAry : A B C 0 B A 0 B
**********Printing colorAry**********
ColorAry : A B C 0 B A 0 B
**********Printing colorAry**********
ColorAry : A B C D B A 0 B
**********Printing colorAry**********

```
ColorAry : A B C D B A   0 B
**********Printing colorAry**********
ColorAry : A B C D B A E B
**********Printing colorAry**********
ColorAry : A B C D B A E B
*********leaving Method1**********
```

debugFile from Data2 method2:

```
********Printing HashTable*******
hashTable [1]->8->5->3->2->4
hashTable [2]->3->4->1
hashTable [3]->7->5->4->2->1
hashTable [4]->7->6->3->2->1
hashTable [5]->7->3->1
hashTable [6]->8->7->4
hashTable [7]->6->5->4->3
hashTable [8]->6->1
**********Entering Method2******************
lastUsedColor: 65
**********Printing colorAry**********
ColorAry : A   0   0   0   0   0   0   0
lastUsedColor: 66
**********Printing colorAry**********
ColorAry : A B   0   0   0   0   0   0
lastUsedColor: 67
**********Printing colorAry**********
ColorAry : A B C   0   0   0   0   0
lastUsedColor: 68
**********Printing colorAry**********
ColorAry : A B C D   0   0   0   0
**********Printing colorAry**********
ColorAry : A B C D B   0   0   0
**********Printing colorAry**********
ColorAry : A B C D B A   0   0
lastUsedColor: 69
**********Printing colorAry**********
ColorAry : A B C D B A E   0
**********Printing colorAry**********
ColorAry : A B C D B A E B
**********Leaving Method2**********
```

debugFile from Data3 method1:
**********Printing HashTable**********
hashTable [1]->3->9->10->8->4->7
hashTable [2]->10->4->7->6->5
hashTable [3]->9->8->1
hashTable [4]->10->7->2->1
hashTable [5]->6->2
hashTable [6]->10->5->2
hashTable [7]->4->2->1
hashTable [8]->9->3->1
hashTable [9]->10->8->3->1
hashTable [10]->9->6->4->2->1
*******entering method1***********
*************Printing colorAry************
ColorAry : A 0 0 0 0 0 0 0 0 0
*************Printing colorAry************
ColorAry : A A 0 0 0 0 0 0 0 0
*************Printing colorAry************
ColorAry : A A 0 0 0 0 0 0 0 0
*************Printing colorAry************
ColorAry : A A B 0 0 0 0 0 0 0
*************Printing colorAry************
ColorAry : A A B B 0 0 0 0 0 0
*************Printing colorAry************
ColorAry : A A B B B 0 0 0 0 0
*************Printing colorAry************
ColorAry : A A B B B 0 0 0 0 0
*************Printing colorAry************
ColorAry : A A B B B C 0 0 0 0
*************Printing colorAry************
ColorAry : A A B B B C C 0 0 0
*************Printing colorAry************
ColorAry : A A B B B C C C 0 0
*************Printing colorAry************
ColorAry : A A B B B C C C 0 0
*************Printing colorAry************
ColorAry : A A B B B C C C D 0
*************Printing colorAry************
ColorAry : A A B B B C C C D 0
*************Printing colorAry************
ColorAry : A A B B B C C C D E
*************Printing colorAry************

ColorAry : A A B B B C C C D E
*********leaving Method1*************

debugFile from Data3 method2:

********Printing HashTable*******
hashTable [1]->3->9->10->8->4->7
hashTable [2]->10->4->7->6->5
hashTable [3]->9->8->1
hashTable [4]->10->7->2->1
hashTable [5]->6->2
hashTable [6]->10->5->2
hashTable [7]->4->2->1
hashTable [8]->9->3->1
hashTable [9]->10->8->3->1
hashTable [10]->9->6->4->2->1
************Entering Method2******************
lastUsedColor: 65
*************Printing colorAry************
ColorAry : A 0 0 0 0 0 0 0 0 0
*************Printing colorAry************
ColorAry : A A 0 0 0 0 0 0 0 0
lastUsedColor: 66
*************Printing colorAry************
ColorAry : A A B 0 0 0 0 0 0 0
*************Printing colorAry************
ColorAry : A A B B 0 0 0 0 0 0
*************Printing colorAry************
ColorAry : A A B B B 0 0 0 0 0
lastUsedColor: 67
*************Printing colorAry************
ColorAry : A A B B B C 0 0 0 0
*************Printing colorAry************
ColorAry : A A B B B C C 0 0 0
*************Printing colorAry************
ColorAry : A A B B B C C C 0 0
lastUsedColor: 68
*************Printing colorAry************
ColorAry : A A B B B C C C D 0
lastUsedColor: 69
*************Printing colorAry************
ColorAry : A A B B B C C C D E
**********Leaving Method2************