

শাফায়েতের ব্লগ

প্রোগ্রামিং ও অ্যালগরিদম টিউটোরিয়াল

Home

অ্যালগরিদম নিয়ে যত লেখা!

আমার সম্পর্কে...

গ্রাফ থিওরিতে হাতেখড়ি – ৩ (ভ্যারিয়েবলে গ্রাফ স্টোর-২)

📅 ডিসেম্বর ২৭, ২০১০ by শাফায়েত

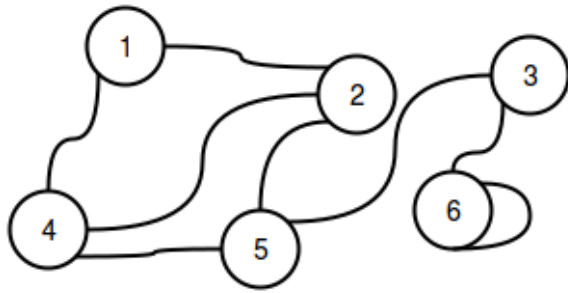


আগের পর্ব

সবগুলো পর্ব

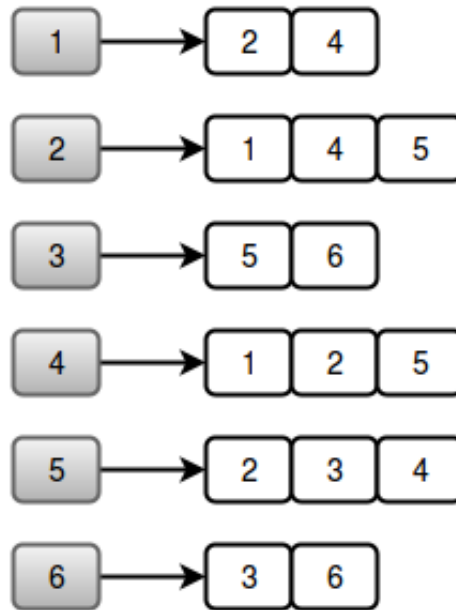
এই পর্বে গ্রাফ স্টোর করার ২য় পদ্ধতি অ্যাডজেসেন্সি লিস্ট নিয়ে লিখব। এ পদ্ধতিতে গ্রাফ স্টোর করে কম মেমরি ব্যবহার করে আরো efficient কোড লেখা যায়। এ ক্ষেত্রে আমরা ডায়নামিক্যালি মেমরি অ্যালোকেট করব, ভয়ের কিছু নেই সি++ এর standard template library(STL) ব্যবহার করে খুব সহজে কাজটা করা যায়। আগের লেখার শেষের দিকে STL এর উপর কয়েকটি টিউটোরিয়ালের লিংক দিয়েছি, আশা করছি ভেক্টর কিভাবে কাজ করে এখন তুমি জানো।

অ্যাডজেসেন্সি লিস্ট শুনতে যতটা ভয়ংকর শুনায়, ব্যাপারটি আসলে তেমনই সহজ। আমরা আবার আগের পোস্টের ছবিটিতে ফিরে যাই:



Nodes	1	2	3	4	5	6
1	0	1	0	1	0	0
2	1	0	0	1	1	0
3	0	0	0	0	1	1
4	1	1	0	0	1	0
5	0	1	1	1	0	0
6	0	0	1	0	0	1

এবার বাজার করার লিস্টের মত একটি লিস্ট বানাই:



এটাই অ্যাডজেসেন্সি লিস্ট, কোন নোডের সাথে কোন নোড যুক্ত আছে সেটার একটা তালিকা। কিন্ত কোড করার সময় কিভাবে এই লিস্টটা স্টোর করবো?

প্রথম উপায়(অ্যারে):

সাধারণ ২ডি অ্যারে ব্যবহার করে লিস্টটি স্টোর করা যায়। যেমন:

```
arr[1][1]=2, arr[1][2]=4;
```

```
arr[2][1]=1; arr[2][2]=4, arr[2][3]=5;
```

কিন্তু এভাবে স্টোর করলে কিছু সমস্যা আছে:

সমস্যা ১. আমাদের ৬টি নোড আছে। প্রতি নোডের সাথে সর্বোচ্চ ৬টি নোড যুক্ত থাকতে পারে(ধরে নিচ্ছি দুটি

নোডের মধ্যে ১টির বেশি সংযোগ থাকবেনা)। এ ক্ষেত্রে আমাদের লাগবে [6][6] আকারের ইন্টিজার অ্যারে। যদি ১ নম্বর নোডের সাথে মাত্র ২টি নোড যুক্ত থাকে তাহলে বাকি array[1][0], array[1][1] কাজে লাগবে, array[1][2] থেকে array[1][6] পর্যন্ত জায়গা কোনো কাজেই লাগবেনা। মনে হতে পারে এ আর এমন কি সমস্যা। কিন্ত চিন্তা কর ১০০০০ টি নোড আছে এমন একটি গ্রাফের কথা। [10000][10000] integer অ্যারে তুমি ব্যবহার করতে পারবেনা, memory limit অতিক্রম করে যাবে, এছাড়া এভাবে মেমরি অপচয় করা ভালো প্রোগ্রামারের লক্ষণ নয় :)। অ্যাডজেসেন্সি ম্যাট্রিক্স ব্যবহার করার সময় যেমন মেমরির সমস্যা হয়েছিলো, এখনও সেই সমস্যা রয়ে যাবে।

সমস্যা ২. অ্যারের কোন ইনডেক্সে কয়টি এলিমেন্ট আছে তার হিসাব রাখতে প্রতি ইনডেক্সের জন্য আরেকটি ভ্যারিয়েবল মেইনটেইন করতে হবে।

দ্বিতীয় উপায়(ভেক্টর):

সমস্যাগুলো দূর করতে আমরা STL vector বা list ব্যবহার করে গ্রাফ স্টোর করব। ভেক্টর/লিস্ট তোমাকে লিস্টের সাইজ বলে দিতে হবেনা, খালি সর্বোচ্চ নোড সংখ্যা বলে দিলেই হবে। এই টিউটোরিয়ালে আমি ভেক্টর ব্যবহার করব কারণ list এ বেশ কিছু সমস্যা আছে।

১০০০০০ নোডের গ্রাফ ইনপুট দেয়ার সময় কখনো ম্যাট্রিক্স হিসাবে দিবেনা, তাহলে ইনপুটের আকারই মাত্রাতিরিক্ত বিশাল হয়ে যাবে। আগের পোস্টে ২য় উদাহরণে যেভাবে দেখিয়েছি সেরকম ইনপুট দিতে পারে, অর্থাৎ প্রথমে নোড আর এজ সংখ্যা বলে দিয়ে তারপর কোন নোডের সাথে কে যুক্ত আছে বলে দিবে। উপরের গ্রাফের জন্য ইনপুট:

“ 6 8 //node-edge

1 2 //node1-node2

1 4

2 4

2 5

4 5

5 3

3 6

6 6

এটি ভেক্টর দিয়ে ইনপুট নিব এভাবে:

```
1 #include<stdio>
2 #include<vector>
3 using namespace std;
4 #define MAX 100000 //maximum node
5 vector<int>edges[MAX];
6 vector<int>cost[MAX]; //parallel vector to store costs;
7 int main()
8 {
9     int N,E,i;
10
11     scanf("%d%d",&N,&E);
12     for(i=1;i<=E;i++)
13     {
14         int x,y;
15         scanf("%d%d",&x,&y);
16         edges[x].push_back(y);
17         edges[y].push_back(x);
18         cost[x].push_back(1);
19         cost[y].push_back(1);
20     }
21     return 0;
22 }
```

cost নামক ভেক্টরটি এ গ্রাফের ক্ষেত্রে দরকার ছিলনা,তবে ওয়েটেড গ্রাফে অবশ্যই দরকার হবে। নিশ্চয়ই বুঝতে পারছ edge ও cost ভেক্টর দুটি সমান্তরাল ভাবে কাজ করবে,অর্থাৎ edge ভেক্টরের যে পজিশনে তুমি দুটি নির্দিষ্ট নোডের কানেকশন পাবে cost ভেক্টরের সেই পজিশনেই তুমি cost পাবে।

যদি ১০০০ বা তার কম নোড থাকে তাহলে ম্যাট্রিক্স বা লিস্ট কোনো ক্ষেত্রেই মেমরি সমস্যা হবেনা। তাও আমরা ভেক্টর দিয়েই গ্রাফ স্টোর করব। কারণ, চিন্তা কর তোমাকে ১০০টা নোডের ম্যাট্রিক্সে ১ এর সাথে কি কি সংযুক্ত আছে বের করতে matrix[1][0],matrix[1][1].....matrix[1][99] এভাবে ১০০টি পজিশন চেক করে কোনটায় কোনটায় ০ নেই বের করতে হবে,১ নম্বর নোডের সাথে যতটি নোডই সংযুক্ত থাকুকনা কেন। তাই এখানেও অগ্যারে আমাদের বাড়তি সুবিধা দিতে পারছেন।

একটা নোডের সাথে কোন কোন নোড যুক্ত আছে বের করা:

ধরো তুমি ১ নম্বর নোডের সাথে যুক্ত সবগুলো নোডের নম্বর চাও, তুমি তাহলে edges[1] এর সাইজ পর্যন্ত লুপ চালাবে এভাবে:

```
“ 1 size=edges[1].size();
   2 for(int i=0; i < size ; i++)
   3 printf("%d ",edges[1][i]);
```

১ >> ০ ১ ০ ০ ০ ১ ১ ০ পুরোটা ঘুরে আসার থেকে ১ >> ২,৬,৭ ঘুরে আসতে কম সময় লাগবে,তাই নয়কি? 😊 ।

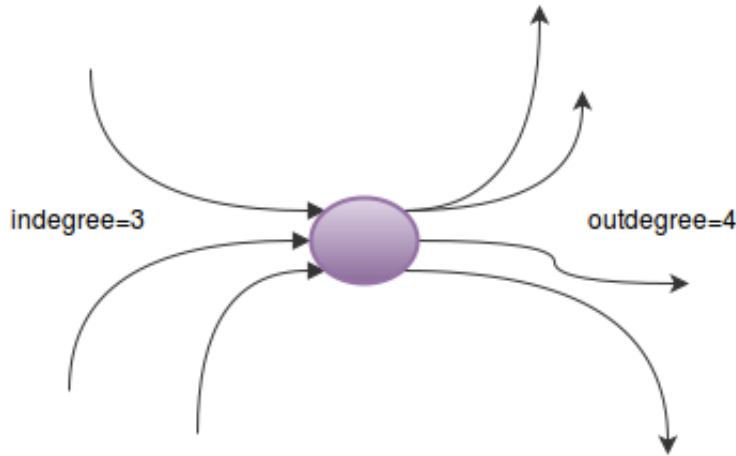
অ্যাডজেসেন্সি ম্যাট্রিক্স কখন লিস্ট অপেক্ষা সুবিধাজনক?

যদি কোনো প্রবলেমে তোমার u, v নোডের এর মধ্যে কোনো এজ আছে নাকি চেক করতে বলে তখন লিস্ট ব্যবহার করলে তোমাকে লুপ চালিয়ে চেক করতে হবে, কিন্ত ম্যাট্রিক্সে জাস্ট $matrix[u][v]$ ইনডেক্স চেক করেই বলে দিতে পারবে তাদের মধ্যে কানেকশন আছে নাকি।

এক্সারসাইজ:

এ পর্যন্ত বুঝে থাকলে তুমি মোটামুটি bfs,dfs এর মত বেসিক অ্যালগোরিদম গুলো শেখার জন্য প্রস্তুত। পরবর্তি লেখাটি পড়ার আগে একটি ছোট exercise করে ফেল। এমন একটি কোড লিখ যেটায় উপরের মত করে ইনপুট দিলে নিচের কাজগুলো করে:

১. একটি adjacency list তৈরি করে। (গ্রাফটিকে directed ধরে নিবে,bidirectional নয়)
২. কোন নোডের সাথে কয়টা নোড যুক্ত আছে,নোডগুলো কি কি সেগুলো প্রিন্ট করে।
৩. indegree হলো একটি নোডে কয়টি নোড প্রবেশ করেছে,outdegree হলো ঠিক তার উল্টোটা। প্রতিটি নোডের outdegree ও indegree প্রিন্ট কর।



পর্ব-৪,বিএফএস:<http://www.shafaetsplanet.com/planetcoding/?p=604>

ফেসবুকে মন্তব্য

19 comments

13 Comments

Sort by Oldest



Add a comment...



Murad Sarkar ·

Bangladesh University of Engineering and Technology (BUET)

VAI I AM GETTING WRONGANSWER IN

<http://www.shafaetsplanet.com/planetcoding/?p=211>

UVA=762.....<http://ideone.com/O76vi> I CANT UNDERSTAND
PLZ HELP.

Like · Reply · 9 August 2012 02:40



Tariq Hasan Anjan ·

Shahjalal University of Science and Technology

খুবই ভালো, আমি এই সিরিজ পড়েই গ্রাফ থিওরি প্রথম শিখেছি/
শিখছি 😊

Like · Reply · 6 · 13 December 2012 15:07



Animesh Kar ·

Software Engineer at Bit Makers Ltd.

Like · Reply · 23 August 2013 11:23



Nasif Ahmed ·

Charigram S.A Khan High School

graph theorem sikhte gele ki age vector stl ba r o kisu sikhe
nite hobe naki..

Like · Reply · 3 · 7 September 2013 23:47



Masuk Sarker

amara same question 😞

Like · Reply · 7 February 2014 03:07



Shawon Ashraf ·

Campus Lead at Microsoft Student Partners Bangladesh

না লাগে না । অ্যারে দিয়েও করা যায়। তবে ইফিশিয়েন্ট
কোড লেখার জন্য ভালো।

Like · Reply · 2 · 9 January 2015 02:07



Mohd. Asfaq-E-Azam Rifat

ভাইয়া, কোডটা করলাম ।। পারলে দেখবেন ।। ধন্যবাদ ...

<http://ideone.com/Afb6at>

Like · Reply · 4 · 29 January 2015 05:31



Jahirul Islam Monir ·

Works at Ahsanullah University of Science & Technology

You code is super.

Like · Reply · 1 · 17 June 2016 17:51



Syed Mohidul Islam ·

United International University

ভাইয়া আমি ভেক্টর array এর সাইজ ১০৫ নিতে পারছি না আমি

code blocks use করার । তাহলে program crash করে ।

Like · Reply · 20 December 2015 22:28



Programming And Problem Solving

Globally অর্থাৎ main() এর বাইরে ডিক্লেয়ার করেন।

Like · Reply · 1 · 3 July 2016 21:53



Piyush Kanti Das ·

Works at Maw lana Bhashani Science and Technology University(MBSTU).

arr[1][1]=2, arr[1][2]=4;

arr[2][1]=1; arr[2][2]=4, arr[2][3]=5;

ভাইয়া এই পার্ট টা বুঝি নাই

Like · Reply · 14 February 2016 21:24



Arafat Kamal Tamzid ·

SEC Affiliated with Shajalal University of Science and Technology

list er elements gula ke array te serially store
korse(2,4,1,4,5,5,6,1.....)

Like · Reply · 29 June 2016 16:20



Pranta Sarker ·

Member at NEUB CSE Society

my solution

Code: <http://paste.ubuntu.com/17276788/>

i will be very glad if you do a verdict to see my solution 😊

thanks in advance 😊

Like · Reply · 1 · 13 June 2016 05:02



Pranta Sarker ·

Member at NEUB CSE Society

this is my solution

Code: <http://paste.ubuntu.com/17276788/>

I will be very glad if you give a verdict to see this 😊 .

thanks in advance 😊

Like · Reply · 1 · 13 June 2016 05:04 · Edited



Arafat Kamal Tamzid ·

SEC Affiliated with Shajalal University of Science and Technology

How can I count indegree and outdegree? can anyone please help..

Like · Reply · 29 June 2016 16:40

Load 3 more comments

 Facebook Comments PluginPowered by **Facebook Comments**

📁 Posted in অ্যালগোরিদম/প্রবলেম সলভিং, প্রোগ্রামিং, সি/সি++ ? Tagged গ্রাফ থিওরি, প্রোগ্রামিং, সি++

32,300 বার পড়া হয়েছে

◀ গ্রাফ থিওরিতে হাতেখড়ি – ২ (ভ্যারিয়েবলে গ্রাফ স্টোর-১)

ওয়ার্ডপ্রেসে ইংরেজি সংখ্যা ও মাসকে বাংলায় রূপান্তর ▶

27 thoughts on "গ্রাফ থিওরিতে হাতেখড়ি – ৩ (ভ্যারিয়েবলে গ্রাফ স্টোর-২)"

Pingback: [গ্রাফ থিওরিতে হাতেখড়ি – ৩ \(ভ্যারিয়েবলে গ্রাফ স্টোর-২\) | ACMSolver - Bangla](#)

**আশিস**

ফেব্রুয়ারি ২৫, ২০১১ at ১২:৫৬ pm

ভাল লাগল। এত লেখিস কি করে! 😊

Reply

**Azad**

জুনে ২০, ২০১১ at ২:৪১ pm

thanks ditay baddho hoilam 😊

Reply

Pingback: [প্রোগ্রামিং এর শৈল্পিক জগৎ-২ | ACMSolver - Bangla](#)

Pingback: [প্রোগ্রামিং এর শৈল্পিক জগৎ-৩](#) | ACMSolver - Bangla



সিয়াম

আগস্ট ৩১, ২০১১ at ১২:৫৫ pm

ভাইয়া প্রবলেম দিলেন সলভ দিলেন না...

আর কিছু ইনপুট আউটপুট দেন...?

স্টেট করার জন্য..?

Reply



শাফায়েত

আগস্ট ৩১, ২০১১ at ১:০৬ pm

@সিয়াম:

এখানেতো ওরকম কোনো সমস্যা দেয়া হয়নি। ইনপুট নিজের ইচ্ছামত ফরমেটে নেন। এই গ্রাফটি ইনপুট নিতে পারেন:

<http://gezeiten.org/public/tikz/graphviz-digraph.png>

টিউটোরিয়ালে যেভাবে নিয়েছে সে ফরমেটেই ইনপুট নিতে পারেন। তারপর কোন নোডের সাথে কি কি যুক্ত আছে প্রিন্ট করেন,যেমন:

1 >> 2,5,8

2 >> 3

.....

.....

6 >> 3,7,8

তারপর indegree,outdegree,যেমন:

1>> in 0 out 3

2>> in 2 out 1

.....

.....

এভাবে গ্রাফ ইনপুট নেয়া এবং traverse করা শেখা হবে,পরবর্তীতে অ্যালগো শিখতে কাজে লাগবে।

Reply



সিয়াম

সেপ্টেম্বর ১৯, ২০১১ at ৪:৫৯ am

```
#include
#include
#include
#include
# define MAX 10000
using namespace std;
int main()
{
freopen("in.txt","r",stdin);
unsigned e,n,i,j;
vector edge[MAX];
vector cost[MAX];
map in,out;
scanf("%u %u",&n,&e);
for(i=0;i<e;i++)
{
int x,y;
scanf("%d %d",&x,&y);
edge[x].push_back(y);
cost[x].push_back(1);
}printf("Connected :\n");
for(j=1;j<=n;j++)
{
printf("%d :",j);
for(i=0; i < edge[j].size() ; i++)
{
in[edge[j][i]]++;
printf("%d ",edge[j][i]);
```

```
}  
cout<<endl;  
out[j]=edge[j].size();  
}  
printf("Indegree Outdegree: \n");  
for(j=1;j> %d out<< %d\n",j,in[j],out[j]);  
return 0;  
}
```

ভাইয়া দেখেন সব ঠিক মত আছে কিনা কোন সাজেশন থাকলে দিয়েন...

Reply



শাফায়েত

সেপ্টেম্বর ২২, ২০১১ at ১:১৯ am

1

ট্যাগের মধ্যে কোডটা পেস্ট করলে ভালো হতো, সুন্দর ফরমেটে আসতো।
যাই হোক কোডটা দেখলাম। ঠিকই আছে মনে হচ্ছে। এখন পরের টিউটোরিয়াল খুলো বিএফএসটা শিখে
ফেলো :)। আর বুঝতে কোনো সমস্যা হলে জানাবে।

Reply

Pingback: [গ্রাফ থিওরিতে হাতেখড়ি-৪ \(ব্রেড ফার্স্ট সার্চ\) | ACMSolver - Bangla](#)

Pingback: [গ্রাফ থিওরিতে হাতেখড়ি-৫\(কোডিং বিএফএস\) | ACMSolver - Bangla](#)

Pingback: [গ্রাফ থিওরিতে হাতেখড়ি ৬: মিনিমাম স্প্যানিং ট্রি | ACMSolver - Bangla](#)

**সঞ্জয় দেবনাথ**

অক্টোবর ৮, ২০১২ at ১২:৩২ pm

ভাইয়া vector দিয়ে গ্রাফ ইনপুট নেয়ার জন্য আপনার প্রদত্ত কোড টা রান করানোর পর ইনপুট না নিয়ে ই স্টপ হয়ে যায় 😞 কোন ভুল তো খুজে পাচ্ছি না।

Reply

**সঞ্জয় দেবনাথ**

অক্টোবর ৮, ২০১২ at ১:০৮ pm

sorry /// হয়েছে 😊

ভেক্টর টা কে গ্লোবাল ডিক্লেয়ার করার পর হয়েছে

Reply

**শাফায়েত**

অক্টোবর ৮, ২০১২ at ৩:৪০ pm

ভ্যারিয়েবলের আকার অনেক বড় হলে গ্লোবাল ডিক্লেয়ার করা উচিত, আমি কোডে গ্লোবাল বানিয়ে দিচ্ছি, ধন্যবাদ।

Reply

**MMHS**

ডিসেম্বর ৩১, ২০১২ at ৮:২৫ am

ভাইয়া, এই code টার কি মোটামুটি efficiency আছে ??

<http://pastebin.com/ABVwrZct>

input এই রকম ছিলঃ

8 10

1 2

2 3

3 4

4 2

1 5

5 6

6 7

6 3

6 8

1 8

(আপনার দেওয়া ছবির input অনুযায়ী)

comment করলে উপকৃত হইতাম ...

Reply

**Nakib**

অক্টোবর ২০, ২০১৩ at ৯:৫১ pm

ভাইয়া , ভেকটর এ আমার একটা সমস্যা হয়

#define 10000 MAX

vector v[MAX]

.....

v[0].push_back(34);

এভাবে pus_back করলে নিচের error দেখায়

errore: 'v' does not name a type

Reply

**Nakib**

অক্টোবর ২১, ২০১৩ at ৭:১৪ am

স্যরি ভাইয়া উপরের কোড ভুল ছিল

```
#define MAX 10000  
vector v[MAX]  
  
.....  
  
v[0].push_back(34);
```

Reply



Arindam Pal

নভেম্বর ২, ২০১৩ at ৩:২৬ pm

what is meant by edge[x]? is it the x-th location of the array[or vector] edges[MAX]?
if so, how do you call size of edge[x]?

Reply



Asif Mujtaba

ডিসেম্বর ১২, ২০১৩ at ১১:৩০ am

ভাইয়া,প্রোগ্রামিং এ আমি একেবারেই বিগেনার,আমার জন্য সহজ কিছু আছে? বিগেনারদের জন্য কোন পরামর্শ?

Reply



শাফায়েত

ডিসেম্বর ২৬, ২০১৩ at ১১:৪৪ pm

আমার ব্লগে যা যা আছে সবই বিগিনারদের জন্যই :)। গ্রাফ থিওরির সহজ প্রবলেমগুলো সলভ করতে পারো। আর কোডিং স্কিল বাড়াতে বেশি করে অ্যাডহকও সলভ করতে হবে।

Reply

Pingback: [গ্রাফ থিওরি এবং একটি রূপকথার গল্প](#) | Shipu's Blog



Soummyo Avik

মে ১১, ২০১৫ at ৭:০২ pm

```
#include
#define MAX 100000+1
#define S(a) scanf("%d",&a)
#define S2(a,b) scanf("%d %d", &a, &b)
#define P1(a) printf("%d",a)
#define P2(a,b) printf("%d %d",a,b)

using namespace std;

vectorgraph[MAX];
vectorcost[MAX];
int in[MAX];
int out[MAX];

int main()
{
    int node, edge;
    S2(node,edge);
    for(int i=1; i<=edge; i++)
    {
        int x, y;
        S2(x,y);
        graph[x].push_back(y);
        cost[x].push_back(1);
    }
    printf("Connected Nodes Are :\n");
    for(int i=1; i ");
    for(int j=0; j<size; j++)
    {
        out[i]++;
        in[graph[i][j]]++;
        P1(graph[i][j]);
        printf(" ");
    }

    printf("Total is : %d\n",size);
}
for(int i=1; i Indegree : %d Outdegree %d\n",i,in[i],out[i]);
}
return 0;
}
```

দেখেন তো ভাইয়া একটু কষ্ট করে কোডটা ওকে আছে কিনা ??

Reply

Pingback: [Prims Algorithm Implementation | saadnoorsalehin](#)

Pingback: [শাফায়েতের ব্লগ » Blog Archive](#)

Pingback: [শাফায়েতের ব্লগ » Blog Archive](#)



Ridowan Ahmed

নভেম্বর ৪, ২০১৬ at ৯:২৪ am

My Code <http://ideone.com/ZP983S>

Reply

Leave a Reply

Connect with:

Powered by [OneAll Social Login](#)

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Website

3 × =
twenty seven



Post Comment

phonetic

probhat

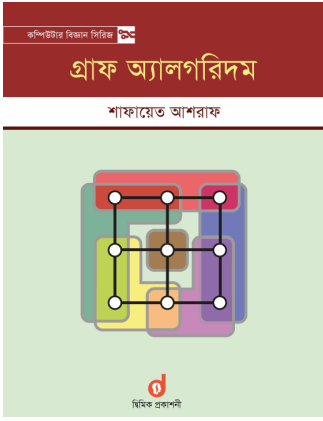
english

সাবস্কাইব

Powered by [OneAll Social Login](#)

আমার সম্পর্কে

শাফায়েত, সফটওয়্যার ইঞ্জিনিয়ার @ **HACKER RANK** (বিস্তারিত...)



Like Share 2.7k

প্রোগ্রামিং কনটেন্ট এবং অ্যালগোরিদম

অনুপ্রেরণা(৩):

কেন আমি প্রোগ্রামিং শিখবো?

কম্পিউটার বিজ্ঞান কেন পড়বো?

প্রোগ্রামিং কনটেন্ট এবং অনলাইন জাজে হাতেখড়ি

অ্যালগোরিদম বেসিক(৬):

বিগ "O" নোটেশন

কমপ্লেক্সিটি ক্লাস(P-NP, টুরিং মেশিন ইত্যাদি)

হাল্টিং প্রবলেম(নতুন)

বাইনারি সার্চ - ১

বাইনারি সার্চ - ২(বাইসেকশন)

ফ্লয়েড সাইকেল ফাইন্ডিং অ্যালগোরিদম

ডাটা স্ট্রাকচার(১১):

লিংকড লিস্ট

স্ট্যাক

কিউ+সার্কুলার কিউ(নতুন)

স্লাইডিং রেঞ্জ মিনিমাম কুয়েরি (ডিকিউ)

ডিসজয়েন্ট সেট(ইউনিয়ন ফাইন্ডিং)

ট্রাই(প্রিফিক্স ট্রি/রেডিক্স ট্রি)

সেগমেন্ট ট্রি-১

সেগমেন্ট ট্রি-২(লেজি প্রপাগেশন)

অ্যারে কমপ্রেসন/ম্যাপিং
লোয়েস্ট কমন অ্যানসেস্টর
বাইনারি ইনডেক্সড ট্রি

গ্রাফ থিওরি(১৮):

গ্রাফ থিওরিতে হাতেখড়ি
অ্যাডজেসেন্সি ম্যাট্রিক্স
অ্যাডজেসেন্সি লিস্ট
ব্রেথড ফার্স্ট সার্চ (বিএফএস)
মিনিমাম স্প্যানিং ট্রি ১ (প্রিমস অ্যালগোরিদম)
মিনিমাম স্প্যানিং ট্রি ২ (করুসকাল অ্যালগোরিদম)
টপোলজিকাল সর্ট
ডেপথ ফার্স্ট সার্চ এবং আবারো টপোলজিকাল সর্ট
ডায়াক্সট্রা
ফ্লয়েড ওয়ার্শল
বেলম্যান ফোর্ড
আর্টিকুলেশন পয়েন্ট এবং ব্রিজ
স্ট্রংলি কানেক্টেড কম্পোনেন্ট
ম্যাক্সিমাম ফ্লো-১
ম্যাক্সিমাম ফ্লো-২
স্টেবল ম্যারেজ প্রবলেম
মিনিমাম ভারটেক্স কভার
ট্রি এর ডায়ামিটার নির্ণয়
লংগেস্ট পাথ প্রবলেম(নতুন)

অ্যালগোরিদম গেম থিওরি(৩):

গেম থিওরি-১
গেম থিওরি-২
গেম থিওরি-৩

ডাইনামিক প্রোগ্রামিং(৮):

শুরুর কথা
ডিপি 'স্টেট', NcR, o-১ ন্যাপস্যাক

কয়েন চেঞ্জ, রক ক্লাইম্বিং
ডিপি সলিউশন প্রিন্ট করা এবং LIS
বিটমাস্ক ডিপি
মিনিমাম ভারটেক্স কভার(গ্রাফ+ডিপি)
লংগেস্ট কমন সাবসিকোয়েন্স(LCS)
ম্যাট্রিক্স চেইন মাল্টিপ্লিকেশন

ব্যাকট্র্যাকিং(১):

ব্যাকট্র্যাকিং বেসিক এবং পারমুটেশন জেনারেটর

নাম্বার থিওরি/গণিত(৪):

মডুলার অ্যারিথমেটিক
প্রাইম জেনারেটর (Sieve of Eratosthenes)
বিটওয়াইজ সিভ
ডিরেকশন

স্ট্রিং ম্যাচিং(১):

রবিন-কার্প স্ট্রিং ম্যাচিং(নতুন)

অন্যান্য(৩) :

ডিরেকশন অ্যারে
মিট ইন দ্যা মিডল
কোয়ান্টাম কম্পিউটার(২)
কোয়ান্টাম কম্পিউটার কী?
কোয়ান্টাম কম্পিউটারের শক্তি এবং সীমাবদ্ধতা

AccessPress Staple | WordPress Theme: AccessPress Staple by AccessPress Themes

