

Day2模拟赛

一、题目概览

中文题目名称	多项式时间哈密顿回路	AI高考数学134分	自动化leetcode
英文题目名称	hamiltonian	math	leetcode
可执行文件名	hamiltonian	math	leetcode
输入文件名	hamiltonian.in	math.in	leetcode.in
输出文件名	hamiltonian.out	math.out	leetcode.out
时间限制	2s	2s	4s
空间限制	512MB	512MB	2GB
测试点数目	25	子任务	子任务
测试点分值	4	子任务	子任务
题目类型	传统	传统	传统
比较方式	全文比较	全文比较	全文比较
是否有部分分	是	是	是

二、注意事项：

- 1.文件名（程序名和输入输出文件名）必须使用小写。
- 2.C/C++中函数main()的返回值类型必须是int，程序正常结束时的返回值必须是0。
- 3.评测环境为Windows，使用lemon进行评测。
- 4.选手不得使用SSH等命令。
- 5.选手不得使用内嵌汇编，#pragma等指令。
- 6.编译选项有 -O2 -std=c++11

多项式时间哈密顿回路

题目描述

这是一道多项式时间哈密顿回路题。

Hamiltonian Cycle Problem is in P

Aimin Hou¹

Abstract

In this paper we present the first deterministic polynomial time algorithm for determining the existence of a Hamiltonian cycle and finding a Hamiltonian cycle in general graphs. Our algorithm can also resolve the Hamiltonian path problem in the traceable graphs.

To achieve the results presented in this paper we construct the corresponding path hologram transformed from the original graph and compute the path set of every vertex in the path hologram with dynamic programming. The path hologram is a multi-segment graph with the vertex $\langle u, k \rangle$ where u is a vertex in the original graph and k is the segment layer of u in the path hologram. The path set of the vertex $\langle u, k \rangle$ is a collection of segment sets consisting of all the vertices located on the same segment layer among all the longest basic paths from the initial vertex S or any vertex on segment j , where $1 \leq j \leq k-1$, to the vertex $\langle u, k \rangle$ in the path hologram. To ensure each path stored in the path set is legal and each segment set of the path set contains only valid vertices, the key strategy of our method is the "consecutive" deleting-replenishing operations recursively on the left/right action field of a vertex, respectively. The greatest contribution of our method is the concept of a path set in the path hologram in which all the legal paths can be stored in $O(n^2)$ space for any a graph, including a complete graph of order n , and any a legal path can be found in deterministic polynomial time.

The space complexity of our algorithm is $O(n^4)$. With the parallel computing, the space complexity can be improved to $O(n^3)$. The time complexity of our algorithm are theoretically $O(n^5 \times d^2)$ on average and $O(n^6 \times d^2)$ in the worst case respectively, where d is the maximum degree of vertex. With the parallel computing, the time complexity can be improved to $O(n^3 \times d^2)$ on average and $O(n^4 \times d^2)$ in the worst case. However, the instances of graphs of order 5 assert that the time complexity of our algorithm are practically $O(n^4 \times d^2)$ on average and in the worst case, respectively. If it is proven to be true that the "While (Flag2)" loop runs in $O(1)$ steps, then with the parallel computing we can design an algorithm with the time complexity of $O(n^2 \times d^2)$ on average and in the worst case, respectively. It is no surprise that we have obtained the time complexity of $O(n^5)$ in the worst case for a complete graph of order n . With the parallel computing, we can process the complete graph of order n with the time complexity of $O(n^3)$ in the worst case.

The path hologram proposed in this paper is only to help readers understand the idea of our method. In fact, our algorithm can be directly applied to the original graph. Besides, our algorithm can deal with the finite general graphs including undirected, directed, and mixed. As a result, the well-known problem HCP in NPC can be now resolved practically in deterministic polynomial time for general graphs in the worst case.

(本人看不懂这种东西，只是感觉很前沿！)

既然哈密顿回路已经属于 P 了，下面这道显然属于 P 的题需要给您来解决：

哈密顿回路是一个人。

哈密顿回路一共做了 n 种 "QwQ", 第 i 种 "QwQ" 有 a_i 个。

她想让最少个数的一种 "QwQ" 的个数最多。

你作为多项式时间复杂度要帮助哈密顿回路。

你可以创造 m 个任意种类的 "QwQ", 并且可以把一些变成另一些。

一个关系 $a \rightarrow b$ 意味着可以把任意个 a 类型的 "QwQ" 变成 b 类型的 "QwQ"。

一个 "QwQ" 可以被转换任意多次, 从 a 类型转换到 b 类型然后转换到 c 类型也是合法的。

因为这是前沿理论, 所以保证对于每个 x , 最多有一个 a 使得 $a \rightarrow x$ 成立。

输入格式

第一行两个数 n, m 。

之后一行, 第 i 个数 fa_i 示第 i 种 "QwQ" 可以被哪种 "QwQ" 转换得到。

如果 fa_i 为 -1 表示这种 "QwQ" 不可以被任何 "QwQ" 转换得到。

之后一行, 第 i 个数 a_i 表示第 i 种 "QwQ" 的个数。

输出格式

输出一行一个数表示答案。

答案即：

你要求让最少个数的一种 "QwQ" 的个数最多的方案, 输出这个方案下最少个数的一种 "QwQ" 的个数。

样例输入1

```
5 5
-1 1 1 1 1
4 5 1 3 2
```

样例输出1

```
3
```

样例输入2

```
10 10
-1 1 1 2 1 5 5 6 10 5
6 1 7 1 7 1 10 5 1 1
```

样例输出2

```
4
```

数据范围

本题不采用子任务评测，每一个测试点为 4 分。

测试点	n	m	a[i]	特殊限制
1	5	5	≤ 5	无
2	10	10	≤ 10	无
3	1000	1000	≤ 1000	特殊限制A
4	1000	1000	≤ 1000	特殊限制A
5	100000	0	无特殊限制	特殊限制A
6	100000	无特殊限制	0	特殊限制A
7	100000	无特殊限制	无特殊限制	特殊限制A
8	1000	1000	≤ 1000	特殊限制B
9	1000	1000	≤ 1000	特殊限制B
10	100000	0	无特殊限制	特殊限制B
11	100000	无特殊限制	≤ 1	特殊限制B
12	100000	无特殊限制	无特殊限制	特殊限制B
13	1000000	无特殊限制	无特殊限制	特殊限制B
14	100000	无特殊限制	无特殊限制	特殊限制C
15	100000	无特殊限制	无特殊限制	特殊限制C
16	1000	0	0	无
17	1000	无特殊限制	0	无
18	1000	0	无特殊限制	无
19	100000	无特殊限制	0	无
20	100000	0	无特殊限制	无
21	100000	无特殊限制	无特殊限制	无
22	200000	无特殊限制	无特殊限制	无
23	500000	无特殊限制	无特殊限制	无
24	800000	无特殊限制	无特殊限制	无
25	1000000	无特殊限制	无特殊限制	无

特殊限制 A: $fa_1 = 1$, 对于 $i \geq 1$ 有 $fa_{i+1} = i$ 。

特殊限制 B: $fa_1 = n$, 对于 $i \geq 1$ 有 $fa_{i+1} = i$ 。

特殊限制 C: 对于 $i \geq 1$ 有 $fa_i = 1$ 。

对于 100% 的数据, $1 \leq n \leq 1000000$, $0 \leq m, a_i \leq 1000000000$ 。

提示

请不要注意常数因子带来的影响。

AI高考数学134分

题目描述

数学高考：两AI考生交卷，最高考到134分

作者：阿霖 吉吉

发布时间：2017-06-07 22:21



芥末堆 阿霖 吉吉 6月7日报道

今天下午，高考数学考试结束后，学霸君研发的高考机器人 Aidam 和准星云学研发的“准星数学高考机器人” AI-MATHS 在北京和成都分别迎战 2017 高考数学科目。

Aidam 挑战全国卷二文科数学卷，取得 134 分的成绩；AI-MATHS 先后向北京高考数学文科卷、全国卷二文科数学卷等多张试卷发起挑战，分别获得 105 分和 100 分的成绩。

3、从算法来说 Aidam 用了多少数据训练，这需要多大的计算量呢？

张凯磊：我们有400台至强十二核 64G 内存的服务器加上1000台左右的辅助服务器，这400台基本都是 Tesla，容量很大。一个Aidam 的背后大概有7个引擎，代数和几何使用的是完全不同的引擎，从算法、逻辑到推理全都不一样，技术上难度很大。

学霸君的首席科学家陈锐锋解释说，机器解题分为三个步骤，第一步是通过句法解构、算式解构和主谓宾解构，转译为机器能看懂的语言，从而让机器读懂题目。第二步是激活神经元之间的连接，机器通过自动证明和推理，得出解题结果，第三步是将解题步骤和结果翻译为人类语言，展示给学生和老师。

这是一道AI高考数学134分题。

你有 400 台 tesla 显卡，现在你需要做一道题来激活神经元之间的连接，从而通过自动证明和推理得出解题结果：
给出三个长度 n 的正整数序列 A, B, C 。

定义 $f(X, l, r)$ 为在序列 X 中，区间 $[l, r]$ 的最大值减去区间 $[l, r]$ 的最小值的差。

定义一个区间 $[l, r]$ 的权值为 $f(A, l, r) \times f(B, l, r) \times f(C, l, r)$ 。

求对于所有 $1 \leq l \leq r \leq n$ ，区间 $[l, r]$ 的权值之和，由于答案可能比较大，所以输出答案对 2^{32} 取模的结果。

输入格式

第一行一个整数 n 。

之后三行，每行 n 个正整数，分别表示 A, B, C 三个序列。

输出格式

一行一个数表示答案。

样例输入

```
5
1 3 5 5 5
2 3 2 1 2
3 5 5 3 5
```

样例输出

```
60
```

数据范围

本题采用子任务评测。

对于 20% 的数据，满足 $1 \leq n \leq 2 \times 10^3$ 。

对于另外 20% 的数据，满足 A, B, C 序列分别单调不减。

对于另外 20% 的数据，满足 A 序列均为 1。

对于 100% 的数据，满足 $1 \leq n \leq 10^5, 1 \leq A_i, B_i, C_i \leq 10^9$ 。

提示

不用注意常数因子带来的影响。

自动化leetcode

题目描述

AI已经会刷LeetCode了



量子位

发布时间: 05-25 13:40 | 万象大会年度获奖创作者, 量子位官方帐号, 财经领域创作者



Gagan Bhatia @gbhatia30 · 5月21日

回复 @DanHendrycks @cHHillee 和 @JacobSteinhardt

I actually worked on a similar data set to produce answers for leetcode questions where my accuracy was 80% at best using a GPT-2 model. This paper seems interesting!

Model	Strict Accuracy			
	Introductory	Interview	Competitive	Average
GPT-2 0.1B	1.00	0.33	0.00	0.40
GPT-2 1.5B	1.30	0.70	0.00	0.68
GPT-Neo 2.7B	3.90	0.57	0.00	1.12
GPT-3 175B	0.20	0.03	0.00	0.06

这是一道自动化 leetcode 题。

人工智能已经以 1.12% 的正确率解决 leetcode 了，作为人类，您需要以 1 的正确率解决自动化 Ynoi，为了自动化 Ynoi，先需要解决这道题：

给定一个长为 n 的序列 a ，每个位置是一个线性变换 $x = |x - a_i|$ ，每次查询给出一个区间 $[l, r]$ 和一个值 v ，依次令 i 从 l 到 r ，访问每个元素 a_i ，将 v 变为 $|v - a_i|$ ，求结束后的 v 的值。

输入格式

第一行两个数 n, m 。

第二行 n 个用空格隔开的数表示序列 a 。

之后 m 行，每行三个用空格隔开的数 l, r, v 表示一次询问。

本题强制在线，所有输入的 l, r, v 均需要异或上次询问的答案，若之前没有询问操作，则为 0。

输出格式

对每个询问操作，输出一行一个数表示答案。

样例输入

```
5 5
4 5 2 5 3
3 5 3
3 3 0
5 0 6
5 0 5
6 0 4
```

样例输出

```
1
4
4
5
1
```

数据范围

本题采用子任务评测

对于 20% 的数据, $1 \leq n, m \leq 1000$ 。

对于另外 10% 的数据, 询问满足 $l = 1, r = n$ 。

对于另外 20% 的数据 $1 \leq a_i, v \leq 100$ 。

对于另外 30% 的数据 $1 \leq a_i, v \leq 10^5$ 。

对于 100% 的数据 $1 \leq n, m \leq 10^5, 1 \leq l, r \leq n, 1 \leq a_i, v \leq 10^7$ 。

说明与提示

第一次询问中, 3 依次经过值为 2, 5, 3 的变化, 变为 1, 4, 1, 答案为 1。

经过解密, 第二次询问为区间 $[2, 2]$, 值为 1。

第二次询问中, 1 依次经过值为 5 的变化, 变为 4, 答案为 4。

经过解密, 第三次询问为区间 $[1, 4]$, 值为 2。

第三次询问中, 2 依次经过值为 4, 5, 2, 5 的变化, 变为 2, 3, 1, 4, 答案为 4。

经过解密, 第四次询问为区间 $[1, 4]$, 值为 1。

第四次询问中, 1 依次经过值为 4, 5, 2, 5 的变化, 变为 3, 2, 0, 5, 答案为 5。

经过解密, 第五次询问为区间 $[3, 5]$, 值为 1。

第五次询问中, 1 依次经过值为 2, 5, 3 的变化, 变为 1, 4, 1, 答案为 1。

提示

不用注意常数因子带来的影响。

