

初赛赛题

1 比赛问题定义

背景：大视频解决方案中，视频业务体验非常关键，视频内容如何有效传送到最终消费者是决定视频体验好坏的核心环节。

本次赛题基本描述

在给定结构的 G 省电信网络中，为了视频内容快速低成本的传送到每个住户小区，需要在这个给定网络结构中选择一些网络节点附近放置视频内容存储服务器。需要解决的问题是：在满足所有的住户小区视频播放需求的基本前提下，如何选择视频内容存储服务器放置位置，使得成本最小。

本次赛题通用性描述

网络结构模型：给定一个由若干网络节点（例如路由器、交换机）构成的网络结构无向图，每个节点至少与另外一个节点通过网络链路相连（网络链路特指两个网络节点之间直接相连的网络通路，中间没有其他网络节点，相当于无向图中的一条边），一个节点可以将收到的数据通过网络链路传输给相连的另一个节点。每条链路的网络总带宽不同（例如某条链路的总带宽为 10Gbps）。而每条链路承载的视频传输需要按照占用带宽的多少收取对应网络租用费，每条链路的单位租用费均不同（例如某条链路的租用费为 1,000 元/Gbps，即 1K/Gbps）。某条链路上被占用的带宽总和不得超过该链路的总带宽。

消费节点：给定的网络结构中有部分网络节点直接连接到小区住户的网络，每个小区住户网络在这个给定的网络结构图中呈现为一个消费节点，不同消费节点的视频带宽消耗需求不同。

视频内容服务器：视频内容服务器存放视频内容（如：电影影片、电视剧等），视频内容服务器的视频数据流可以经由网络节点与链路构成的网络路径流向消费节点，视频内容服务器的输出能力没有上限，可以服务多个消费节点，一个消费节点也可以同时从多台视频内容服务器获取视频流。部署一台视频内容服务器需要费用成本（例如 300,000 元/台，即 300K/台），所有服务器的成本均相同。

比赛程序内容：请你设计一个程序寻找最优的视频内容服务器部署方案：从网络结构模型中选择一部分网络节点，在其上/附近一对一的部署视频内容服务器，视频内容服务器与对应的这个节点直连，与对应的这个网络节点之间的通信没有带宽限制、也没有通信成本。提供的部署方案需要使得视频流从视频内容服务器经过一些网络节点和链路到达消费节点，并满足所有消费节点的视频带宽消耗需求。

在满足所有消费节点视频带宽消耗需求的前提下，使得耗费的总成本（视频内容服务器部署成本+带宽租用成本）最低。部署方案不仅需要包括部署视频内容服务器的节点位置，而且还要包括每个消费节点与所有视频内容服务器之间的网络路径以及路径上占用的带宽。

比赛胜负规则

若给定的网络拓扑模型不存在满足条件的方案，则输出无解，程序运行时间越短者胜出。若存在满足条件的方案，则成本越低者胜出。若两个方案的成本相同，则程序运行时间越短者胜出。

补充说明：

1. 两个网络节点之间最多仅存在一条链路，链路上下行方向的网络总带宽相互独立，并且上下行方向的总带宽与网络租用费相同。例如对于网络节点 A 与 B 之间的链路，该条链路上的总带宽为 10Gbps，单位租用费为 1K/Gbps，则表示 A->B、B->A 两个方向上的网络总带宽分别均为 10Gbps，并且租用费均为 1K/Gbps。如果某条数据流在该链路 A->B 方向的占用带宽为 3Gbps，那么该数据流在该链路的租用费为 3K，并且该链路 A->B 方向的剩余可用带宽为 7Gbps。而 B->A 方向的剩余可用带宽不受该数据流的影响，仍为 10Gbps。

2. 每个网络节点最多仅能连接一个消费节点，每个消费节点仅能连接一个网络节点。消费节点与连接的网络节点之间的链路总带宽无限大，并且网络租用费为零。

3. 网络节点数量不超过 1000 个，每个节点的链路数量不超过 20 条，消费节点的数量不超过 500 个。

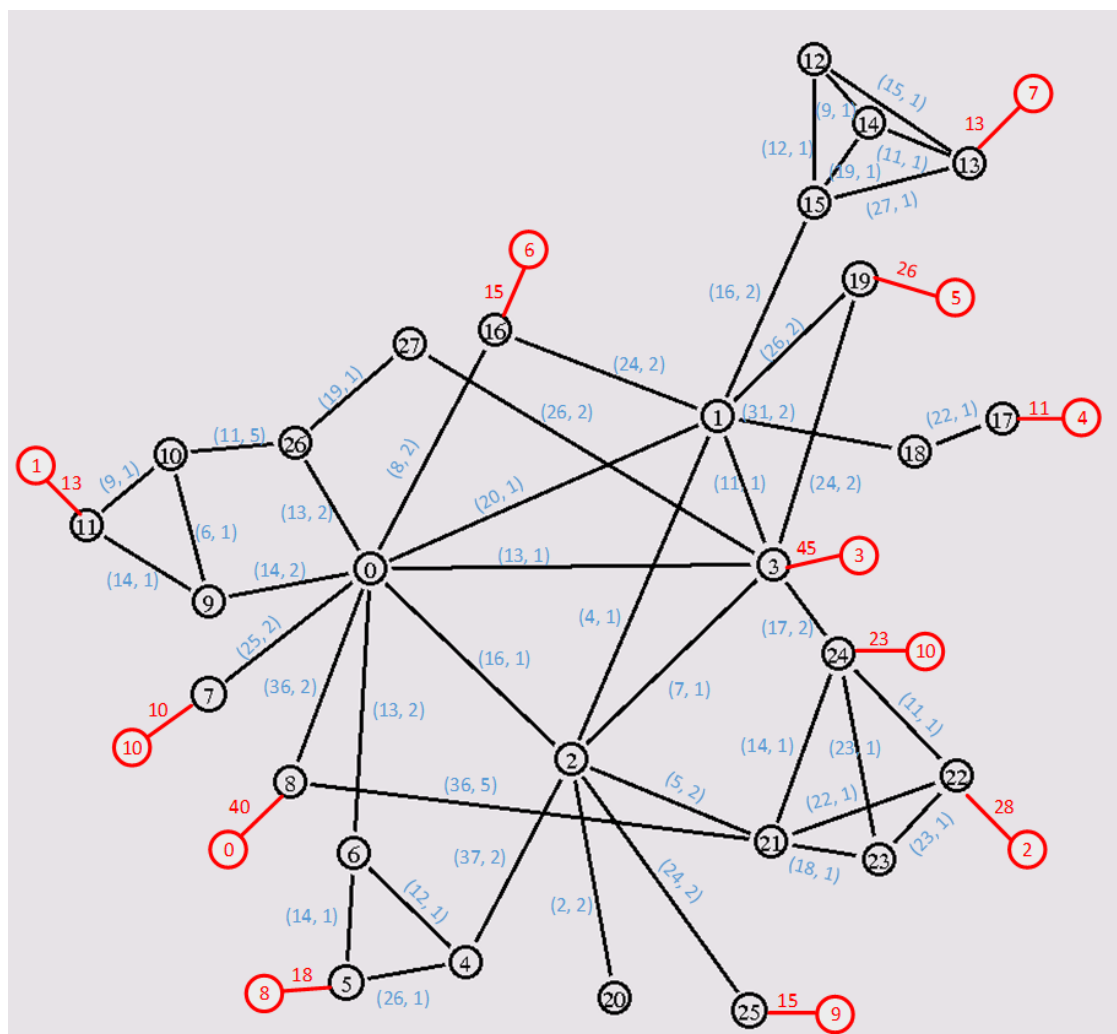
4. 链路总带宽与网络租用费为[0, 100]的整数，视频内容服务器部署成本与消费节点的视频带宽消耗需求为[0,5000]的整数。

5. 部署方案中，网络路径上的占用带宽必须为整数。

6. “满足消费节点的带宽消耗需求”是指输出给消费节点的带宽总和不得小于该消费节点的视频带宽消耗需求。

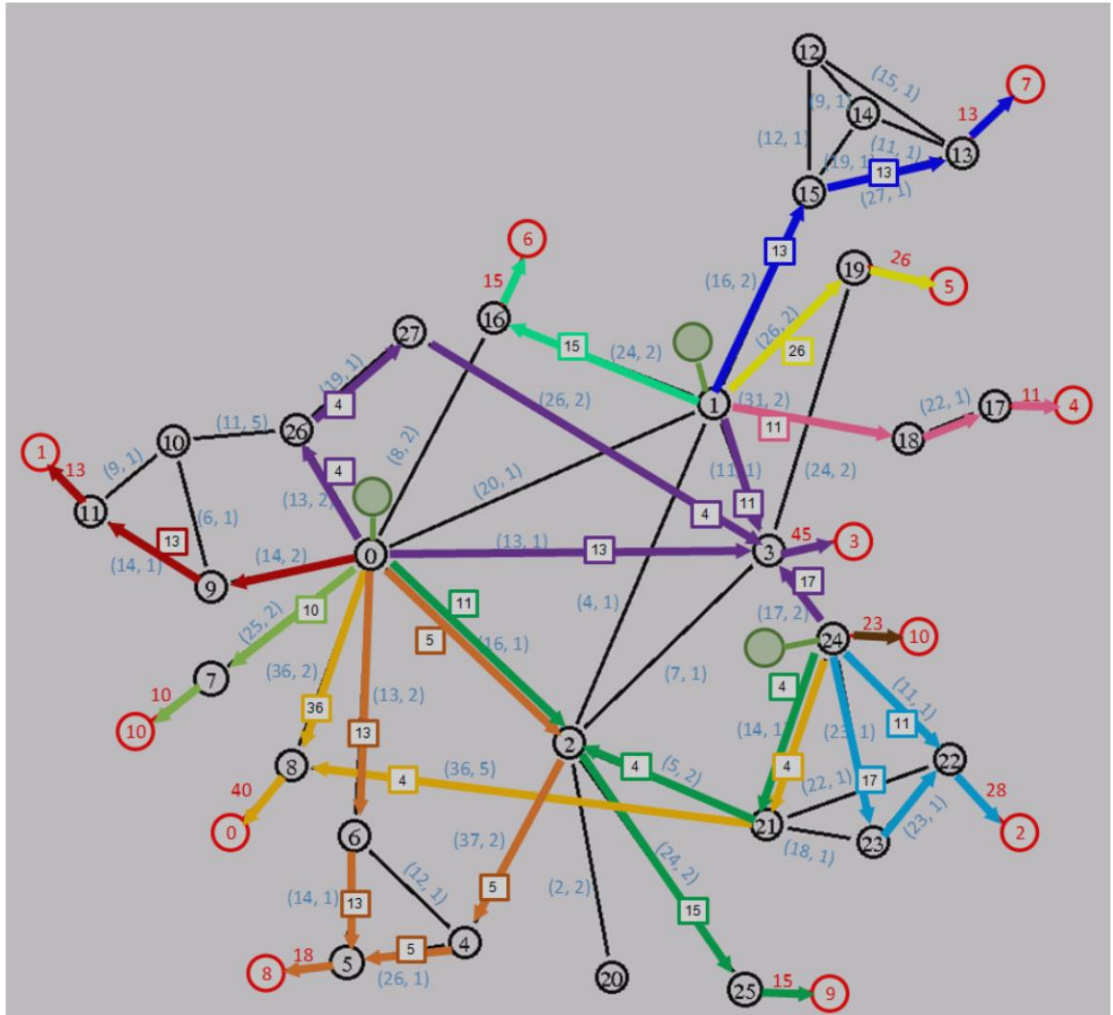
7. 每个网络节点上最多仅可部署一台视频内容服务器。

比赛用例示例：



上图为 G 省网络拓扑图，黑色圆圈为网络节点，红色圆圈为消费节点，圆圈内的数字为节点编号。节点之间的连线为网络链路。链路上的标记(x, y)中，x 表示链路总带宽（单位为 Gbps），y 表示每 Gbps 的网络租用费。消费节点相连链路上的数字为消费节点的带宽消耗需求（单位为 Gbps）。

现在假设需要在该网络上部署视频内容服务器，满足所有消费节点的需求。一个成本较低的方案如下图所示，其中绿色圆圈表示已部署的视频内容服务器，通往不同消费节点的网络路径用不同颜色标识，并附带了占用带宽的大小：



但该方案的成本不一定是最低的。因此现在需要参赛者提供的程序能够针对不同的比赛用例，给出成本最低的部署方案。

2 程序输入与输出

输入文件格式

程序输入为一个以空格分隔的文本文件，文件每行以换行符（ASCII' \n' 即 0x0a）为结尾。

文件格式为：

网络节点数量 网络链路数量 消费节点数量

(空行)

视频内容服务器部署成本

(空行)

链路起始节点 ID 链路终止节点 ID 总带宽大小 网络租用费

..... (如上链路信息若干行)

(空行)

消费节点 ID 相连网络节点 ID 视频带宽消耗需求

..... (如上终端用户信息若干行)

(文件结束)

说明：

1. 网络节点 ID 与消费节点 ID 均为大于或等于 0 的整数。
2. 文本中出现的所有数值均为正整数，数值上限为 $2^{32} - 1$ 。

输入文件示例（参照第一节中的用例）：

28 45 12

100

0 16 8 2

0 26 13 2

0 9 14 2

0 8 36 2

0 7 25 2

0 6 13 2

0 1 20 1

0 2 16 1

0 3 13 1

1 19 26 2

1 18 31 2

1 16 24 2

1 15 16 2

1 2 4 1

1 3 11 1

2 4 37 2

2 25 24 2

2 21 5 2

2 20 2 2

2 3 7 1

3 19 24 2

3 24 17 2

3 27 26 2

4 5 26 1

4 6 12 1

5 6 14 1

8 21 36 5

9 10 6 1

9 11 14 1

10 26 11 5

10 11 9 1

12 13 15 1

12 14 9 1

12 15 12 1

13 14 11 1

13 15 27 1

14 15 19 1

17 18 22 1

21 22 22 1

21 23 18 1

21 24 14 1

22 23 23 1

22 24 11 1

23 24 23 1

26 27 19 1

0 8 40

1 11 13

2 22 28

3 3 45

4 17 11

5 19 26

6 16 15

7 13 13

8 5 18

9 25 15

10 7 10

11 24 23

输出文件格式

程序输出为一个以空格分隔的文本文件，文件每行以换行符（ASCII‘ \n’ 即 0x0a ）为结尾。

文件格式为：

如果不存在满足条件的方案，则输出一行

NA

（ 文件结束 ）

如果存在满足条件的方案，则按如下格式输出：

网络路径数量

（ 空行 ）

网络节点 ID-01 网络节点 ID-02 网络节点 ID-n 消费节点 ID 占用带宽

大小

..... (如上网网络路径信息若干行，每条网络路径由若干网络节点构成，路径的起始节点 ID-01 表示该节点部署了视频内容服务器，终止节点为某个消费节点)

(文件结束)

说明：

1. 网络路径数量不得超过 50000 条。
2. 单条路径的节点数量不得超过 1000 个。
3. 不同网络路径可按任意先后顺序输出。
4. 网络节点 ID 与消费节点 ID 的数值必须与输入文件相符合，如果 ID 数值不存在于输入文件中，则将被视为无效结果。
5. 文本文件中出现的所有数值必须为正整数，数值大小不得超过 232 - 1。

输出文件示例 (参照第一节中的用例部署方案)：

19

0 9 11 1 13

0 7 10 10

0 8 0 36

0 6 5 8 13

0 2 4 5 8 5

0 2 25 9 11

24 21 8 0 4

24 21 2 25 9 4

24 23 22 2 17

24 22 2 11

24 10 23

24 3 3 17

0 3 3 13

0 26 27 3 3 4

1 3 3 11

1 18 17 4 11

1 19 5 26

1 15 13 7 13

1 16 6 15

3 单个用例的评分机制

有解用例的排名机制

按下面流程对参赛者结果进行排名：

Step1：对于提交的结果，进行合法性检验(详见题目描述)；

Step2：程序运行时间不得超过 90s；

若不满足上述的结果则本用例得分为 0；

Step3：计算部署方案的成本，成本越小，排名越优；

Step4：在成本相同的结果里，用程序运行时间进行排名，时间越短，排名越优。

无解用例的排名机制

按下列流程对参赛者结果进行排名：

Step1：对于提交的结果，验证是否识别出该用例无解，若无法识别或者算法运行时间超 90s，则本用例得分为 0；

Step2：用程序的运行时间进行排名，时间越短，排名越优。

单个用例的评分标准如下：

根据上面排名流程得到的排名，使用标准分计分(排名第一的提交者为 100 分)。

若所有人均未得到正确结果，则所有人均得分为 0。

4 最终得分机制

比赛平台会使用 N 个测试用例判题，该 N 个测试用例分为初级、中级、高级三个等级，参赛者对于每个测试用例都会得到一个百分制分数，使用加权平均分(初级权重为 0.2，中级权重为 0.3，高级权重为 0.5)作为该参赛者的最终得分。

前期系统会提供练习用例用于判题与排名，4 月 1 日开始提供正式用例。截止初赛结束前，系统将在用例上自动运行选手最后一次提交的代码并进行评分，所得分数为初赛最终成绩。在比赛初期，比赛平台只提供初级、中级的练习用例，故此时满分为 50 分，在比赛后期，才会提供高级练习用例（具体时间会在网站公告通知），此时满分才为 100。

运行环境

CPU : Intel(R) Xeon(R) CPU E5-2680 V4 @ 2.40GHz

内存 : 2G

内核 : 单核

编译器 : gcc 4.8.4 ; java 1.7.0_95 ;

操作系统 : linux Ubuntu 14.04.4 LTS 64 位 , 内核版本 Linux version 4.2.0-27-generic

SDK : 为方便选手做题 , 分别提供 c++(兼容 c) 和 Java SDK 包供参考 (见赛题下载包) , 详细描述信息请见 SDK 目录下的 readme.txt。

QQ 讨论交流

成渝赛区 : 364374276

杭夏赛区 : 131946455

京津东北赛区 : 271699173

江山赛区 : 545377507

上合赛区 : 221036348

武长赛区 : 132176435

西北赛区 : 569746514

粤港澳赛区 : 479203724

新加坡赛区 : 372796988