

CCF CSP 计算机软件能力认证

CCF CSP

第 38 次认证

时间：2025 年 6 月 8 日 13:30 ~ 17:30

题目名称	正态分布	机器人复健指南	消息解码	月票发行	博物馆
题目类型	传统型	传统型	传统型	传统型	传统型
输入	标准输入	标准输入	标准输入	标准输入	标准输入
输出	标准输出	标准输出	标准输出	标准输出	标准输出
每个测试点时限	1.0 秒	1.0 秒	2.0 秒	4.0 秒	1.0 秒
内存限制	512 MiB	512 MiB	512 MiB	512 MiB	512 MiB
子任务数目	10	20	10	4	4
测试点是否等分	是	是	是	否	否

正态分布 (normal)

【题目描述】

对于正态分布随机变量 $X \sim N(\mu, \sigma^2)$ (均值 μ 、标准差 σ)，查表计算 $P(X \leq n)$ 。具体来说，我们首先需要将 X 转换为标准正态分布 Z ：

$$Z = \frac{X - \mu}{\sigma}$$

那么 $X \leq n$ 的概率也就等于 $Z \leq \frac{n - \mu}{\sigma}$ 的概率：

$$P(X \leq n) = P(Z \leq \frac{n - \mu}{\sigma})$$

而对于服从标准正态分布的 Z ，其小于等于某值的概率 $P(Z \leq m)$ 可以通过查表得出。图 1 展示了 m 取值从 0.00 到 1.49 的结果（步长 0.01），其中每列对应 m 的百分位、每行对应 m 的十分位和整数部分。该表可继续向下延伸，这里只展示部分结果。

$P(Z \leq m)$	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
0.0	0.5000	0.5040	0.5080	0.5120	0.5160	0.5199	0.5239	0.5279	0.5319	0.5359
0.1	0.5398	0.5438	0.5478	0.5517	0.5557	0.5596	0.5636	0.5675	0.5714	0.5753
0.2	0.5793	0.5832	0.5871	0.5910	0.5948	0.5987	0.6026	0.6064	0.6103	0.6141
0.3	0.6179	0.6217	0.6255	0.6293	0.6331	0.6368	0.6406	0.6443	0.6480	0.6517
0.4	0.6554	0.6591	0.6628	0.6664	0.6700	0.6736	0.6772	0.6808	0.6844	0.6879
0.5	0.6915	0.6950	0.6985	0.7019	0.7054	0.7088	0.7123	0.7157	0.7190	0.7224
0.6	0.7257	0.7291	0.7324	0.7357	0.7389	0.7422	0.7454	0.7486	0.7517	0.7549
0.7	0.7580	0.7611	0.7642	0.7673	0.7704	0.7734	0.7764	0.7794	0.7823	0.7852
0.8	0.7881	0.7910	0.7939	0.7967	0.7995	0.8023	0.8051	0.8078	0.8106	0.8133
0.9	0.8159	0.8186	0.8212	0.8238	0.8264	0.8289	0.8315	0.8340	0.8365	0.8389
1.0	0.8413	0.8438	0.8461	0.8485	0.8508	0.8531	0.8554	0.8577	0.8599	0.8621
1.1	0.8643	0.8665	0.8686	0.8708	0.8729	0.8749	0.8770	0.8790	0.8810	0.8830
1.2	0.8849	0.8869	0.8888	0.8907	0.8925	0.8944	0.8962	0.8980	0.8997	0.9015
1.3	0.9032	0.9049	0.9066	0.9082	0.9099	0.9115	0.9131	0.9147	0.9162	0.9177
1.4	0.9192	0.9207	0.9222	0.9236	0.9251	0.9265	0.9279	0.9292	0.9306	0.9319

图 1: 标准正态分布常用值表

如图 2 所示， $Z \leq 1$ 的概率即为阴影部分的面积，查看表中 1.0 对应行、0.00 对应列即可得到近似结果 0.8413。

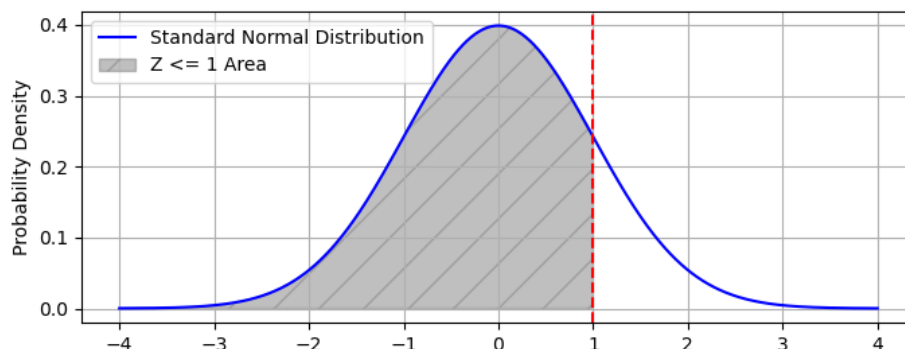


图 2: 标准正态分布示意图

在本题中你需要模拟上述查表的过程，处理 k 个如下查询：对于给定的参数 μ 、 σ 和 n ，计算 $P(X \leq n)$ 的结果在表中的哪一行、哪一列？

其中行列下标均从 1 开始：

- 行：0.0 对应第 1 行，0.1 对应第 2 行，依此类推.....
- 列：0.00, 0.01, \dots , 0.09 依次对应第 1、第 2 到第 10 列。

【输入格式】

从标准输入读入数据。

输入的第一行包含一个正整数 k ，表示查询的个数。

接下来输入 k 行，每行包含空格分隔的三个整数 μ 、 σ 和 n ，表示一个查询。

【输出格式】

输出到标准输出。

每个查询输出一行，包含空格分隔的两个整数 i 和 j ，表示查询的结果位于表中第 i 行、第 j 列。

【样例输入】

```
1 4
2 0 1 1
3 2 10 127
4 2 50 227
5 5 100 350
```

【样例输出】

```
1 11 1
2 126 1
3 46 1
4 35 6
```

【样例解释】

第一个查询等价于计算 $P(Z \leq \frac{1-0}{1})$ ，如题目描述所示，查看表中 1.0 对应行（第 11 行）、0.00 对应列（第 1 列）即可。

【子任务】

全部的数据满足：

- $k \leq 20$;
- 参数 μ 、 σ 和 n 均为整数;
- $0 \leq \mu \leq n \leq 1000$;
- $1 \leq \sigma \leq 100$ 且标准差 σ 是 100 的因子。

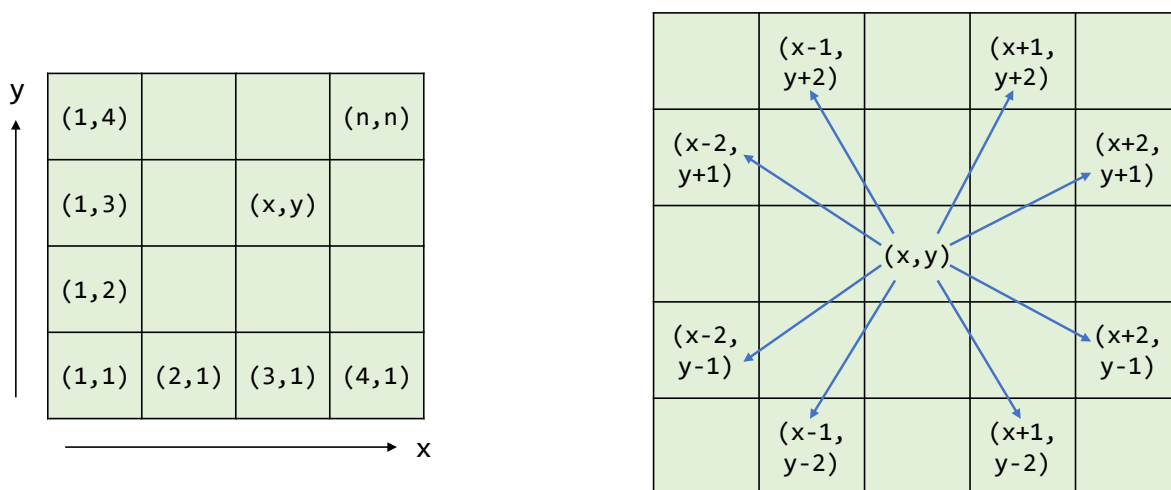
机器人复健指南 (jump)

【题目背景】

西西艾弗岛某山脉深处出土了一台远古机器人，具体年代已不可考。初步修缮后，研究人员尝试操控机器人进行些简单的移动。

【题目描述】

整个实验场地被划分为 $n \times n$ 个方格，从 $(1,1)$ 到 (n,n) 进行编号。机器人只能在这些方格间移动，不能走出场地范围。如下图所示，假设机器人当前位于 (x,y) ，那么接下来可以向周围八个方向跳跃移动（如果目标方格在场地范围内）：



若机器人只能跳动不超过 k 步，场地内有多少方格（包括起始位置）可以抵达？

【输入格式】

从标准输入读入数据。

输入的第一行包含空格分隔的两个正整数 n 和 k ，分别表示场地大小和跳动步数。

输入的第二行包含空格分隔的两个正整数 x 和 y ，表示机器人的起始位置（保证位于场地内）。

【输出格式】

输出到标准输出。

输出一个整数，表示 k 步内可以抵达的方格总数。

【样例 1 输入】

```
1 4 1
2 1 1
```

【样例 1 输出】

```
1 3
```

【样例 2 输入】

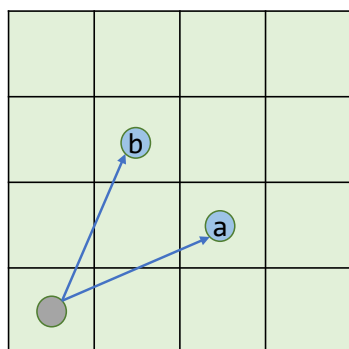
```
1 4 2
2 1 1
```

【样例 2 输出】

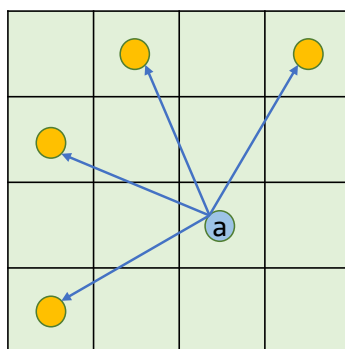
```
1 8
```

【样例 2 解释】

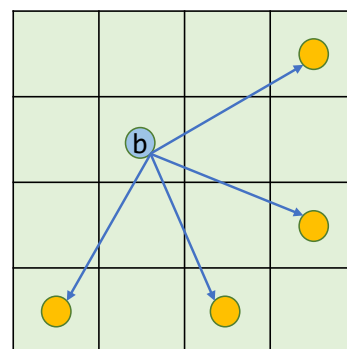
如下图所示，初始位置、第一步和第二步跳跃抵达的位置总计为 8。



第 1 步



第 2 步 - a



第 2 步 - b

【子任务】

80% 的测试数据满足： $k \leq 3$ ；

全部的测试数据满足： n 、 k 均大于 0 且不超过 100。

消息解码 (decodeft8)

【题目背景】

西西艾弗岛上的居民经常使用一种短消息服务来互相通信。这种短消息的特点是，每条消息仅需 72 个比特，可以用来发送些简短的信息。发送的信息包含收发双方的代号和发送方的位置。每位居民都有一个代号，代号包含 1 到 11 个字符，且只能包含大写字母、数字和下划线。特别地，多数居民的代号长 5 到 6 位，并且依次由 1 到 2 位数字或大写字母、1 位数字和 3 位字母组成。符合这样特征的代号被称为**典型代号**。西西艾弗岛上的位置，可以用数字编号 1 至 32400 表示。

代号的数字表示

一个代号可以唯一地用一个数字表示，具体办法是：

1. 将不足 11 位的代号从**结尾**处用空格补充到 11 位；
2. 将每个字符用一个数字表示，数字的计算方法如下：
 - 空格用 0 表示；
 - 数字 0 至 9 用 1 到 10 表示；
 - 大写字母 A 至 Z 用 11 到 36 表示；
 - 下划线 _ 用 37 表示。
3. 将第一个字符的数字乘以 38^{10} ，第二个字符的数字乘以 38^9 ，第三个字符的数字乘以 38^8 ，.....，第十一个字符的数字乘以 38^0 ，然后将这些数相加。

例如，代号 **ABCD200_3** 的数字表示可以计算为： $11 \times 38^{10} + 12 \times 38^9 + 13 \times 38^8 + 14 \times 38^7 + 3 \times 38^6 + 1 \times 38^5 + 1 \times 38^4 + 37 \times 38^3 + 4 \times 38^2 + 0 \times 38^1 + 0 \times 38^0 = 71101055775146872$

典型代号的短数字表示

典型代号除了常规的数字表示，还有一种更简洁的短数字表示：

1. 将 5 位的代号从**开头**处用空格补充到 6 位；
2. 将第一个字符转换为数字：
 - 空格用 0 表示；
 - 数字 0 至 9 用 1 到 10 表示；
 - 大写字母 A 至 Z 用 11 到 36 表示；
3. 将第二个字符转换为数字：
 - 数字 0 至 9 用 0 到 9 表示；
 - 大写字母 A 至 Z 用 10 到 35 表示；
4. 将第三个字符转换为数字：
 - 数字 0 至 9 用 0 到 9 表示；
5. 将第四、五、六个字符转换为数字：
 - 大写字母 A 至 Z 用 0 到 25 表示；

6. 将第一个字符的数字乘以 $36 \times 10 \times 26^3$ ，将第二个字符的数字乘以 10×26^3 ，将第三个字符的数字乘以 26^3 ，将第四个字符的数字乘以 26^2 ，将第五个字符的数字乘以 26^1 ，将第六个字符的数字乘以 26^0 ，然后将这些数相加。

例如，典型代号 **A0BCD** 的短数字表示可以计算为： $\underline{0} \times 36 \times 10 \times 26^3 + \underline{10} \times 10 \times 26^3 + \underline{0} \times 26^3 + \underline{1} \times 26^2 + \underline{2} \times 26^1 + \underline{3} \times 26^0 = 1758331$

代号的散列值

为了能在 72 个比特的消息中表达 11 位的代号，并容纳前述信息，居民们采用的方法是：

- 在第一次通信中使用代号的数字表示，此后可以使用代号的**散列值**作为代替，因为散列值的长度较短；
- 如果是典型代号，也可以直接使用其短数字表示来节省空间。

由代号的数字表示可以唯一地计算出代号的散列值，但由代号的散列值不能唯一地计算出代号的数字表示。对于接收消息的一方而言，如果收到的消息中含有代号的散列值，则可以通过此前收到的代号来推算该散列值所对应的代号。代号的 n 位散列值的计算方法如下：

1. 计算代号的数字表示；
2. 将代号的数字表示乘以 47055833459；
3. 将结果除以 2^{64-n} ，取整数部分；
4. 将结果对 2^n 取模。

例如，代号 **ABCD200_3** 的 25 位散列值可以计算为：

$$71101055775146872 \times 47055833459 = 3345719439314381360096790248$$

$$3345719439314381360096790248 \div 2^{64-25} \bmod 2^{25} = 7770595$$

(典型) 代号 **A0BCD** 的 12 位散列值可以计算为：

$$\text{数字表示 } \underline{11} \times 38^{10} + \underline{1} \times 38^9 + \underline{12} \times 38^8 + \underline{13} \times 38^7 + \underline{14} \times 38^6 = 69279249490478336$$

$$69279249490478336 \times 47055833459 \div 2^{64-12} \bmod 2^{12} = 909$$

【题目描述】

消息可以被分为两种：简单消息和复杂消息。其中复杂消息包含完整的代号的数字表示；而简单消息仅包含典型代号的短数字表示或代号的散列值，同时可以包含发送方所在位置。

简单消息的第一个二进制位是 0，随后的 28 位（位序为高位在前，下同）表示接收方的代号，随后的 28 位表示发送方的代号，接下来的 15 位是发送方的位置编号。表示一方代号的数字有两种情况：如果是代号的散列值，则为代号的 25 位散列值；如果是典型代号的短数字表示，则为该值加上 2^{25} 。位置编号如果为 0 表示该消息**不包括位置编号**，否则表示发送方的位置编号。

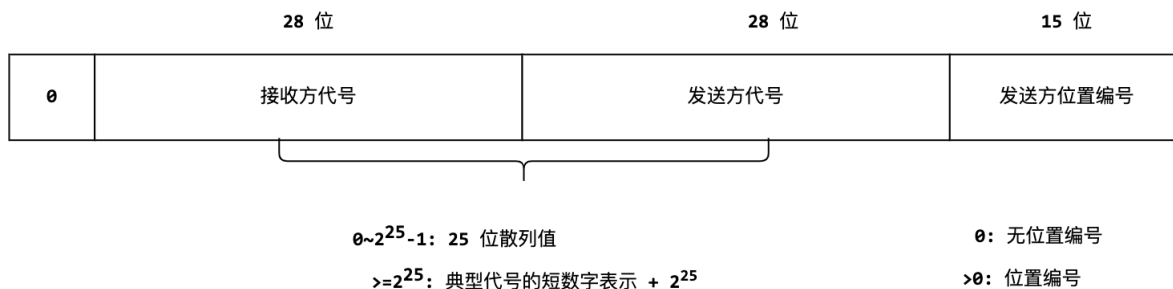


图 3: 简单消息

复杂消息的第一个二进制位是 1，随后的 58 位是一方的代号的数字表示；接下来的 12 位是另一方的代号的 12 位散列值；最后的 1 位表示二者的关系：0 表示首先的一方（即用代号的数字表示的那一方）是发送方，1 表示首先的一方是接收方。



图 4: 复杂消息

一条消息的文字表示包含用空格分隔的三部分，分别是接收方的代号、发送方的代号和发送方的位置编号。如果消息不包括位置编号，则无第三部分。其中，如果一方的代号是用散列值推断的，则在该方的代号前加上井号（#）；如果通过散列值无法推断出该方的代号（可能因为此前的消息没有收到），则用 ### 表示。用散列值推断代号时，使用收到该消息前接收到的消息中出现过的代号推断。用于推断的代号包括出现在简单消息中的用短数字表示的典型代号，和出现在复杂消息中的用数字表示的代号；而不包括此前通过散列值间接推断出的。如果有多个代号的散列值都符合，则使用最后收到消息中出现的代号；如果最后收到的消息中的收发双方代号的散列值都符合，则使用最后收到的消息中发送方的代号。

试编写程序，将给出的消息的比特序列转换为对应的文字表示。

【输入格式】

从标准输入读入数据。

输入的第一行是一个整数 n ，表示消息的数量。接下来的 n 行中，每行包含一个由 72 个 0 或 1 组成的字符串，表示先后收到的 n 条 72 比特的消息。

【输出格式】

输出到标准输出。

输出 n 行，表示这些消息对应的文字表示。

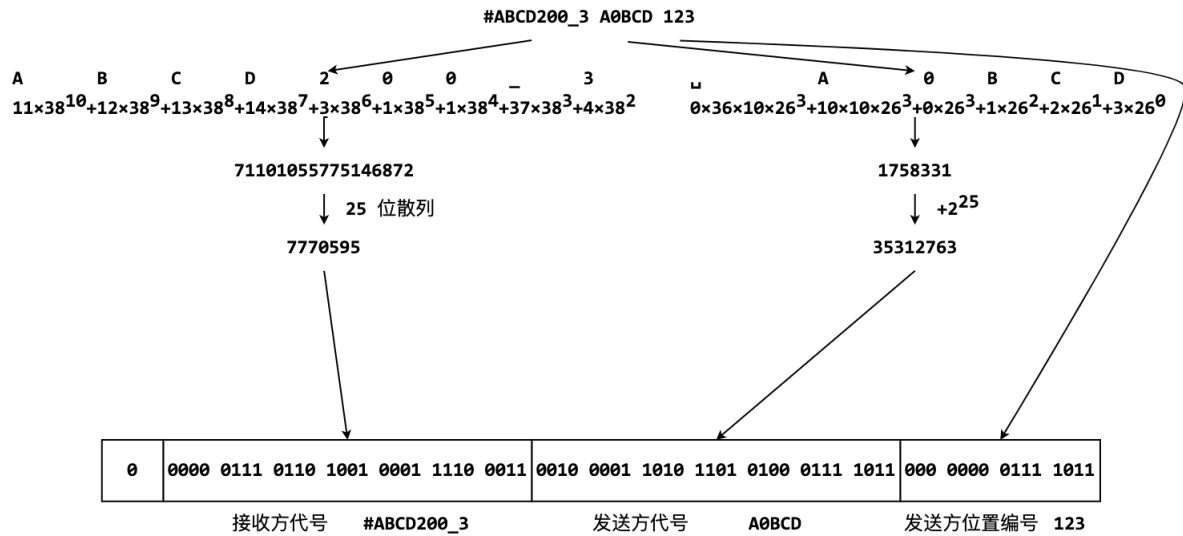


图 5: 简单消息和其文字表示

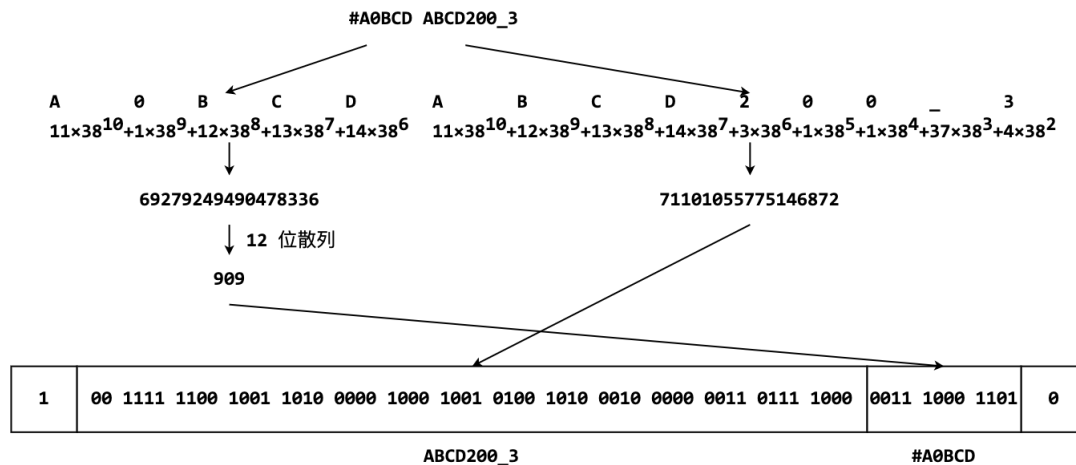


图 6: 复杂消息和其文字表示

【样例 1】

见题目目录下的 *1.in* 与 *1.ans*。

【样例 1 解释】

前两个输入为题目描述中的例子。需要注意的是，在解析第一个输入时，收到的代号散列值尚不能推断出代号，因此用 ### 表示。

第三个输入是复杂消息。根据代号的数字表示值解析第一个代号为 ABCD200_4，第二个代号的 12 位散列值为 001110110100，即 948。此时向前搜索发现曾经出现的 ABCD200_3 的 12 位散列值也是 948，因此可以推断出第二个代号为 ABCD200_3，并加 # 表示这个代号是推断而来的。输入的最后一位是 0，表示第一个代号是发送方，第二个代号是接收方，因此要将 #ABCD200_3 放在前面。

第四个输入也是复杂消息，解析得第一个代号为 **PY4XMCHJZTN**、第二个代号的 12 位散列值为 970。此时在前面出现过的代号中找不到 12 位散列值为 970 的代号，因此无法推断出第二个代号，用 **###** 表示。输入的最后一位是 1，表示第一个代号是接收方，第二个代号是发送方，因此要将 **PY4XMCHJZTN** 放在前面。

第五个输入也是复杂消息，解析得第一个代号为 **ABCD200_4**、第二个代号的 12 位散列值为 948。需要注意的是，曾经出现的代号 **ABCD200_3**、**ABCD200_4**、**PY4XMCHJZTN** 的 12 位散列值都是 948，我们选取最近在第四个输入中收到的 **PY4XMCHJZTN** 作为推断的代号，因此得到本条消息的结果。注意虽然第五个输入和第三个输入完全相同，但是由于代号的 12 位散列值发生了碰撞，因此推断出的代号可能不同。

第六个输入也是复杂消息，解析得第一个代号为 **ABCD200_5**、第二个代号的 12 位散列值为 948。这时，选取第五个输入中收到的 **ABCD200_4** 作为推断的代号，因为它是最近收到的 12 位散列值是 948 的代号。需要注意的是，虽然本条消息中出现的代号 **ABCD200_5** 的 12 位散列值也是 948，但是在根据散列值推断代号时，仅考虑此前收到的消息中出现的代号，而不考虑本条消息中出现的代号。

第七个输入和第六个输入相同，此次选取的是第六个输入中收到的 **ABCD200_5** 作为推断的代号。

【子任务】

测试点	n	额外性质
1, 2, 3	≤ 100	仅含有复杂消息，且散列值与所有出现的代号均不匹配
4, 5, 6		仅含有简单消息，且表示双方代号的数字均不小于 2^{25}
7, 8		无
9, 10	$\leq 10^5$	

全部的数据满足：输入的消息数据一定是按上述编码规则得到的，但可能存在由散列值无法推断出代号的情况。

月票发行 (ticket)

【题目背景】

西西艾弗岛上的西埃斯公园正在准备 5202 年的月票发行工作。

【题目描述】

西埃斯公园发行的月票编号为由小写字母 (a-z) 和井号 (#) 组成的字符串:

- 月票编号的字符串长度为 n , 记作位置 1 到位置 n ;
- 月票编号中有且仅有 m 个位置固定为井号;
- 其余 $n - m$ 个位置可以自由填写 26 种小写字母。

为了显示月票的发行方为西西艾弗岛上的西埃斯公园, 可发行的月票编号必须满足如下要求:

- ccf 和 cspark 两种子串均出现至少一次 (子串为连续的若干字符);
- 至少存在一对 ccf 和 cspark, 满足 ccf 在 cspark 前面 (左侧) 出现。

例如, 当字符串长度为 16、存在一个位置为 10 的井号时, 以下编号的月票均为可发行的:

- ccfcspark#ccfcsp
- ccfcspark#cspar
- csparkccf#cspar
- ccsparccf#cspar

当然, 可发行的月票远不止上面四种。

你的任务就是帮助西埃斯公园计算: 总共能够发行多少张不同的月票? 两张月票不同当且仅当两张月票的编号至少有一处位置的字符不同。

答案可能很大, 你只需求出其对 998244353 取模的结果即可。

【输入格式】

从标准输入读入数据。

输入的第一行包含两个正整数 n 和 m , 表示月票编号的字符串长度和井号的数量。

接下来一行包含 m 个正整数 a_1, a_2, \dots, a_m , 表示指定的井号位置。

【输出格式】

输出到标准输出。

输出一个整数, 表示可发行月票总数对 998244353 取模的结果。

【样例 1 输入】

```
1 12 1
2 11
```

【样例 1 输出】

```
1 2028
```

【样例 1 解释】

在本样例中，受井号限制 `ccf` 和 `cspark` 只能分布在前面 10 个字符中，因此二者各自只能出现一次，具体有以下三种情况：

- `ccf` 填写在 1 ~ 3 的位置，`cspark` 填写在 4 ~ 9 的位置；
- `ccf` 填写在 1 ~ 3 的位置，`cspark` 填写在 5 ~ 10 的位置；
- `ccf` 填写在 2 ~ 4 的位置，`cspark` 填写在 5 ~ 10 的位置。

上述三种情况中，都有另外两个可以填写任意小写字母的空余位置。因此可发行的月票总数为 $3 \times 26^2 = 2028$ 。

【样例 2 输入】

```
1 14 2
2 4 8
```

【样例 2 输出】

```
1 35151
```

【样例 2 解释】

在本样例中，`cspark` 只能填写在 9 ~ 14 的位置，此时 `ccf` 只要填写在 1 ~ 3 或者 5 ~ 7 任意一处即可。

将 `ccf` 填写在一处后，另外一处的三个位置便可以填写任意小写字母；但请注意，两处均填写 `ccf` 的月票会被重复计算一次。

因此可发行的月票总数为 $2 \times 26^3 - 1 = 35151$ 。

【样例 3 输入】

```
1 16 1
2 10
```

【样例 3 输出】

```
1 474661742
```

【子任务】

本题采用捆绑测试，你只有通过一个子任务中的所有测试点才能得到该子任务的分数。

- 子任务一（30 分）：保证 $n \leq 15$ ；
- 子任务二（30 分）：保证 $n \leq 10^5$ ；
- 子任务三（30 分）：保证 $n \leq 10^9, m \leq 10^4$ ；
- 子任务四（10 分）：无特殊限制。

对于所有数据，保证 $9 \leq n \leq 10^9, 1 \leq m \leq \min(n, 10^6), 1 \leq a_1 < a_2 < \cdots < a_m \leq n$ 。

【提示】

可能存在无可发行月票的情况，此时的计算结果为 0。但是对所有测试点皆输出 0 不会得分。

博物馆 (museum)

【题目背景】

西西艾弗岛是历史悠久的旅游胜地，岛上有很多的博物馆。

【题目描述】

小 C 和小 F 二人正在毕业旅行。他们在深夜抵达了西西艾弗岛。

西西艾弗岛的交通系统由 n 个路口和连接它们的 m 条无向道路组成，每个路口都有一家旅馆可供住宿，小 C 和小 F 今夜可以选择任意一个路口的旅馆入住。

西西艾弗岛上有大量的博物馆，小 C 计划第二天去其中一些博物馆游览。他在网上收集了 q 份不同的一日游攻略，其中第 i 份攻略包含 c_i 个博物馆，分别位于第 $a_{i,1}, a_{i,2}, \dots, a_{i,c_i}$ 个路口。

正在二人畅想明日美好的游玩体验时，小 F 忽然想起一件事：明天西西艾弗岛的市政施工队会在恰好一个路口处施工，也就是说，明天一整天二人都无法经过该路口。这让今晚住宿地点的选择变得非常重要。具体来说，如果第 x 个路口明天施工，小 C 和小 F 今晚住在第 u 个路口，那么对于一座在第 v 个路口的博物馆，二人能游览它当且仅当存在一条从 u 到 v 的路径不经过 x 。特别地，如果 $x = u$ 或 $x = v$ ，那么这座博物馆是无法被游览的。

小 C 和小 F 都是旅行特种兵，只要选定了一份攻略，他们就会游览这份攻略里所有能游览的博物馆。不过小 F 忘记了明天到底是哪个路口要施工了，他决定把所有可能的情况都推演一遍。对于一份攻略，他规定 $f(x)$ 表示在第 x 个路口明天施工的情况下，两人今晚按照最优策略选择旅馆，明天最多能游览的博物馆数量。

现在小 F 希望对所有 q 份攻略都求出 $\sum_{i=1}^n f(i)$ 。由于小 F 计算能力比较差，所以他 把地图和攻略都发给了你，希望你帮他算出来。

【输入格式】

从标准输入读入数据。

输入的第一行包含三个正整数 n, m, q ，分别表示路口的数量，双向道路的数量和攻略的数量。

之后 m 行每行包含两个正整数 u, v ，表示一条路口 u 和路口 v 之间的双向道路。

之后 q 行每行表示一份攻略，第 i 行开头一个正整数 c_i ，随后紧跟 c_i 个两两不同的正整数 $a_{i,1}, a_{i,2}, \dots, a_{i,c_i}$ ，含义见上。

【输出格式】

输出到标准输出。

输出 q 行，每行一个正整数，第 i 行表示对于第 i 份攻略， $\sum_{j=1}^n f(j)$ 的值。

【样例 1 输入】

```
1 7 9 1
2 1 2
3 1 3
4 2 3
5 1 4
6 1 5
7 4 5
8 1 6
9 1 7
10 6 7
11 3 2 3 4
```

【样例 1 输出】

```
1 17
```

【样例 1 解释】

$f(1) = f(2) = f(3) = f(4) = 2$ 。

对于其他的路口都有 $f(i) = 3$ 。

【子任务】

本题采用捆绑测试，你只有通过一个子任务中的所有测试点才能得到该子任务的分数。

- 子任务一（15 分）：保证 $n, q \leq 100$ ；
- 子任务二（15 分）：保证所有的路口和道路组成一条链；
- 子任务三（30 分）：保证 $m = n - 1$ ；
- 子任务四（40 分）：无特殊限制。

对于所有数据，保证 $1 \leq n, q \leq 10^5, m \leq 2 \times 10^5, \sum_{i=1}^q c_i \leq 3 \times 10^5$ 。

保证在一开始所有的路口之间两两可达，两个路口之间至多只有一条道路，且不存在一条道路的起点与终点相同。即所有的路口与道路构成一张简单无向连通图。

保证每份攻略中博物馆所处的路口两两不同。