

类型推断 (Deduction)

题目背景

小 X 在学习写解释器，他遇到了一些难以克服的问题，希望你能帮助他获得答案。

你需要注意的是：出题者**不**希望你使用 `__int128`。

题目描述

小 X 要创造一门叫做 X++ 的强类型语言，这意味着所有变量的类型必须要在编译期确定。小 X 希望你能帮助他确定所有变量的类型。

X++ 的语法规则如下：

- X++ 中的**基本**变量类型为 `int`，`long`，`char`。其中，`int` 可以存储 $-(2^{31} - 1) \sim 2^{31} - 1$ 的整数，`long` 可以存储 $-(2^{63} - 1) \sim 2^{63} - 1$ 的整数，`char` 可以存储**一个**字符，且只能是 ASCII 中的可见字符。类型为基本变量类型的变量为基本变量。他的声明为 `type xxx = yyy`，其中 `type` 为基本变量类型，`xxx` 为变量名，`yyy` 为基本变量类型字面量或同类型的变量名。
- X++ 中的代码可以用 `;` 分割成多条语句。若 `;` 在字符串字面量里，则不视作分隔符，单条语句必为声明语句。程序的最后一个非空白字符必须为 `;`。
- X++ 中存在**复合变量**，变量类型为 `comb`。复合变量是一个列表，它由多个基本变量或复合变量构成，他们均被称为复合变量的**成员**。它的声明为 `comb xxx = {.....}`，`comb` 为固定关键字，`xxx` 为变量名，`.....` 为多个基本类型字面量，或其他任意已声明的变量名。
- X++ 的变量存在**层级**。定义基本变量的层级为 0，复合变量的层级等于其所有成员变量层级的最大值加一。
- X++ 的字面量存在**自动推导**。对于一个字面量，X++ 会先后尝试着将它声明为 `int`，`long`，`char` 和 `comb`。
- X++ 的程序中，所有的作为运算符的位置，`,``=` 和 `;` 前后可以加空格，也可以不加空格。所有可以加空格的地方，空格的个数可以任意多，例如，`int a=1;` 和 `int a = 1;` 是相同的意思，X++ 应当忽略不必要的空格。
- X++ 所有声明变量的语句，变量类型可以写 `auto` 以让 X++ 进行自主推导。
- X++ 不应当把换行符视作分割语句的条件，而是应当视作一个空格。

X++ 的声明可能会出现非法声明：

- X++ 中，只允许声明上面提到的几种变量。如果尝试声明其他种类的变量，应当报出错误 `TypeError: Type not supported.` 并退出解释程序。
- X++ 的自动推导中，如果所有类型均不能满足要求，则会报出错误 `ValueError: Cannot declare variable.` 并退出解释程序。
- X++ 中，同样的变量名反复声明是不允许的，即使两次声明的为同一类型也一样。这时，你应该报出错误 `NameError: Name has been used.` 并退出解释程序。
- X++ 中，若显式的指明了变量的类型，则会优先考虑变量类型是否可满足，例如 `long a = 1;` 是一条合法语句，即使 X++ 的类型推导会认为 `1` 是一个 `int`，最后它依旧会是一个 `long` 类型的数据。
- X++ 中声明的变量和实际类型不相容时，报出错误 `ValueError: Cannot declare variable.` 并退出解释程序。
- X++ 的基本变量可能会出现非法声明，会出现在所声明的变量不属于任何基本变量类型的情况，报出错误 `ValueError: Cannot declare variable.` 并退出解释程序。特别的，若输入的整数字面量 $\geq 2^{63}$ ，你要输出 `ValueError: Signed intenger overflow.`；若尝试声明了一个长度不为 1 的字符字面量，应该报出错误

ValueError: Type does not match. 并退出解释程序；若尝试声明了一个非 ASCII 可见字符的字符字面量，应该报出错误 ValueError: Type does not match. 并退出解释程序。若尝试声明的 int 或 long 型变量，除了第一个字符外仍有字符 - ，应该报出错误 ValueError: Type does not match. 并退出解释程序。

- X++ 的复合变量可能会出现非法声明，可以发现会出现在 { 和 } 不匹配，两个 , 之间没有变量的情况，此时报出错误 SyntaxError: Symbols does not match. 并退出解释程序。
- X++ 程序的最后一个非空白字符若不为 ; , 此时报出错误 SyntaxError: Symbols does not match. 并退出解释程序。
- X++ 的变量声明只允许出现大写或小写英文字母，如果违反此条，报出错误 NameError: Invalid variable name. 并退出解释程序。

如果程序中存在多个错误，报出按照上述描述顺序的第一个错误。如果存在多个同类型错误，报出程序中第一个出现的错误。

以上所有未尽之处，均和 C++ 中定义相同。

小 X 想让你找出所有声明的变量并写出其变量类型。对于 comb 类型，你还要输出它的层级。

注：报出错误指的是把错误信息输出到标准输出中。

输入格式

输入有多组数据。

第一行一个整数 T ，代表 T 组数据。

对于每组数据，第一行一个整数 n ，代表 X++ 源码长度。

接下来自第二行第一个字符起 n 个字符，代表 X++ 源码。保证最后一个字符紧接着的字符为换行符。

输出格式

对于每组数据：

若输入的源码为合法的 X++ 程序，按照所声明的变量名的字典序，一行一个输出它的类型。特别的，对于 comb 类型，你需要在该行多输出一个整数代表它的层级，和其类型相隔恰好一个空格。

否则，输出错误信息。

样例 #1

样例输入 #1

```
5
11
auto a = 1;
35
auto a = 111111111111111111111111111111;
22
auto a = 1;auto b={a};
10
auto b = {
20
auto b = {1, 2,, 3};
```

样例输出 #1

```
a int
ValueError: Signed integer overflow.
a int
b comb 1
SyntaxError: Symbols does not match.
SyntaxError: Symbols does not match.
```

样例 #2

样例输入 #2

```
3
28
auto x = {1, 22, 3, {4, 5}};
47
auto x = 184032;
int z = "d";
long
v = 0923748;
41
char x = "X";auto XXX = "{";comb x = {1};
```

样例输出 #2

```
x comb 2
ValueError: Type does not match.
NameError: Name has been used.
```

样例 #3

样例输入 #3

```
2
22
int a = 6468465658676;
25
long b = 940182938904123;
```

样例输出 #3

```
ValueError: Type does not match.
b long
```

提示

对于所有数据, $T \leq 5, n \leq 5000$ 。

存在 60% 的数据, $T = 1$ 。

存在 50% 的数据, 若源程序非空, 则源程序必有错误。

存在 20% 的数据, 源程序中只有一种变量类型。

不保证以上的数据类型各不相交, 但保证总存在一组数据, 使得其只满足以上三个特殊性质中的一种。

特别的, 这道题中我们认为 ASCII 的可见字符只有所有的大小写英文字母, 阿拉伯数字以及 `!"#$%&'()*+,-./:;<=>?@[\]^_{|} ``。