//这道题比较有意思的地方，在于录入时对于cin和cout的卡时，虽然觉得这一行为没多大必要，但是还是是得注意以后若cin和cout要调用多次的时候得注意卡时。

#include<iostream>

#include<string>

#include<algorithm>

#include<queue>

#include<vector>

#include<sstream>

#include<stack>

using namespace std;

struct student {

string id;

string name;

int score;

};

vector<student>all\_student;

int comp1(student a,student b)

{

if (a.id < b.id)

return 1;

else

return 0;

}

int comp2(student a, student b)

{

if (a.name < b.name)

return 1;

else if (a.name == b.name)

{

if (a.id < b.id)

return 1;

else

return 0;

}

else

return 0;

}

int comp3(student a, student b)

{

if (a.score < b.score)

return 1;

else if (a.score == b.score)

{

if (a.id < b.id)

return 1;

else

return 0;

}

else

return 0;

}

int main()

{

int num, col;

cin >> num;

cin >> col;

char id[12],name[12];

int score;

for (int i = 0; i < num; i++)

{

student temp;

scanf("%s %s %d", id, name, &score);

temp.id = id;

temp.name = name;

temp.score = score;

all\_student.push\_back(temp);

}

if (col == 1)

{

sort(all\_student.begin(), all\_student.end(), comp1);

}

else if (col == 2)

{

sort(all\_student.begin(), all\_student.end(), comp2);

}

else if (col == 3)

{

sort(all\_student.begin(), all\_student.end(), comp3);

}

for (int i = 0; i < num; i++)

{

printf("%s %s %d\n", all\_student[i].id.c\_str(), all\_student[i].name.c\_str(), all\_student[i].score);

}

}