一、题目说明

题目非常简洁15. 3Sum，读懂题目后，理解不难。

但 实话说，我们提交代码后，Time Limit Exceeded，最主要的是给了非常长的测试用例，我本地运行后87秒，确实时间非常长。

二、我的解答及问题

下面是我的解答代码，通过该例子可以学习如何写测试代码：

```
#include<iostream>
#include<vector>
#include<unordered_map>
#include<algorithm>
using namespace std;
class Solution{
    public:
        inline bool checkExist(vector<int>&nums,vector< vector<int> > &res){
            vector<int> vec_tmp;
            for(vector< vector<int>>::iterator
iter=res.begin();iter!=res.end();++iter){
                vec_tmp = *iter;
                if(nums[0]==vec_tmp[0] && nums[1]==vec_tmp[1])
                    return true;
            }
            return false;
        }
        vector< vector<int> > threeSum(vector<int>&nums){
            vector< vector<int> > result;
            if(nums.size() < 3)   return result;
            unordered_map<int,int>ump;
            sort(nums.begin(), nums.end());
            vector<int> vec_tmp;

            for(int i=0;i<nums.size();i++){
                ump[nums[i]] = i;
            }

            for(int i=0;i<nums.size()-2;i++){
                for(int j=i+1;j<nums.size()-1;j++){
                    const int gap = (0 - nums[i] - nums[j]);
                    if(ump.find(gap) != ump.end() && ump[gap]>j){
                        vec_tmp = {nums[i],nums[j],gap};
                        if(!checkExist(vec_tmp, result)){
                            result.push_back(vec_tmp);
                        }
                    }
                }
            }

            return result;
        }
};
int main(){
    Solution s;
```

```
    vector<int> v =  {-1, 0, 1, 2, -1, -4};
    vector< vector<int> > r = s.threeSum(v);
    vector<int> vec_tmp;
    for(vector< vector<int>>::iterator iter=r.begin();iter!=r.end();++iter){
        vec_tmp = *iter;
        for(vector<int>::iterator it=vec_tmp.begin();it!=vec_tmp.end();++it){
            cout<<*it<<" ";
        }
        cout<<endl;
    }
    return 0;
}
```

这个里面，我用了twoSum的思路，用了unordered_map查找加速。但还是超时了，最主要的原因在于本题目用了双重循环。

经过多次优化，终于正确了，不过性能太差：

```
Runtime: 608 ms, faster than 5.03% of C++ online submissions for 3Sum.
Memory Usage: 21.1 MB, less than 8.83% of C++ online submissions for 3Sum.
```

```
#include<iostream>
#include<vector>
#include<unordered_map>
#include<algorithm>
using namespace std;
class Solution{
    public:
        vector< vector<int> > threeSum(vector<int>&nums){
            vector< vector<int> > result;
            if(nums.size() < 3)   return result;
            unordered_map<int,int>ump;
            unordered_map<string,int> uniq;
            sort(nums.begin(), nums.end());

            for(int i=0;i<nums.size();i++){
                ump[nums[i]] = i;
            }

            for(int i=0;i<nums.size()-2;i++){
                if(i>0 && nums[i]==nums[i-1])    continue;
                for(int j=i+1;j<nums.size()-1;j++){
                    const int gap = (0 - nums[i] - nums[j]);
                    if(ump.find(gap) != ump.end() && ump[gap]>j){
                        string str=to_string(nums[i]) + to_string(nums[j])+
to_string(gap);
                        if(uniq.find(str) == uniq.end() || uniq[str]!=1){
                            result.push_back({nums[i],nums[j],gap});
                            uniq[str] = 1;
                        }
                    }
                }
            }
```

```
            return result;
        }
};
int main(){
    Solution s;
    vector<int> v =  {-1, 0, 1, 2, -1, -4};
//  vector<int> v =  {0, 0, 0, 0};

    vector< vector<int> > r = s.threeSum(v);
    vector<int> vec_tmp;
    for(vector< vector<int>>::iterator iter=r.begin();iter!=r.end();++iter){
        vec_tmp = *iter;
        for(vector<int>::iterator it=vec_tmp.begin();it!=vec_tmp.end();++it){
            cout<<*it<<" ";
        }
        cout<<endl;
    }
    return 0;
}
```

三、正确的解答

网上找了一个性能还不错的实现，思路就是用left和right指示位置，然后求和。

```
#include<iostream>
#include<vector>
#include<unordered_map>
#include<algorithm>
using namespace std;
class Solution{
    public:
    vector<vector<int>> threeSum(vector<int>& nums) {
        vector<vector<int>> ans;
        if(nums.size() < 3)   return ans;
        sort(nums.begin(), nums.end());
        for(int i=0; i < nums.size()-2; i++){
            if(i>0 && nums[i]==nums[i-1])    continue;
            int target = -nums[i], l = i+1, r = nums.size()-1;
            while(l < r){
                if(nums[l] + nums[r] < target){
                    l++;
                }else if(nums[l] + nums[r] > target){
                    r--;
                }else{
                    ans.push_back({nums[i], nums[l], nums[r]});
                    while(l<r && nums[l]==nums[++l]);
                    while(r>l && nums[r]==nums[--r]);
                }
            }
        }
        return ans;
    }
};

int main(){
    Solution s;
```

```
    vector<int> v =  {-1, 0, 1, 2, -1, -4};
```

```
//  vector<int> v =
{82597,-9243,62390,83030,-97960,-26521,-61011,83390,-38677,12333,75987,46091,837
94,19355,-71037,-6242,-28801,324,1202,-90885,-2989,-95597,-34333,35528,5680,8909
3,-90606,50360,-29393,-27012,53313,65213,99818,-82405,-41661,-3333,-51952,72135,
-1523,26377,74685,96992,92263,15929,5467,-99555,-43348,-41689,-60383,-3990,32165
,65265,-72973,-58372,12741,-48568,-46596,72419,-1859,34153,62937,81310,-61823,-9
6770,-54944,8845,-91184,24208,-29078,31495,65258,14198,85395,70506,-40908,56740,
-12228,-40072,32429,93001,68445,-73927,25731,-91859,-24150,10093,-60271,-81683,-
18126,51055,48189,-6468,25057,81194,-58628,74042,66158,-14452,-49851,-43667,1109
2,39189,-17025,-79173,13606,83172,92647,-59741,19343,-26644,-57607,82908,-20655,
1637,80060,98994,39331,-31274,-61523,91225,-72953,13211,-75116,-98421,-41571,-69
074,99587,39345,42151,-2460,98236,15690,-52507,-95803,-48935,-46492,-45606,-7925
4,-99851,52533,73486,39948,-7240,71815,-585,-96252,90990,-93815,93340,-71848,587
33,-14859,-83082,-75794,-82082,-24871,-15206,91207,-56469,-93618,67131,-8682,757
19,87429,-98757,-7535,-24890,-94160,85003,33928,75538,97456,-66424,-60074,-8527,
-28697,-22308,2246,-70134,-82319,-10184,87081,-34949,-28645,-47352,-83966,-60418
,-15293,-53067,-25921,55172,75064,95859,48049,34311,-86931,-38586,33686,-36714,9
6922,76713,-22165,-80585,-34503,-44516,39217,-28457,47227,-94036,43457,24626,-87
359,26898,-70819,30528,-32397,-69486,84912,-1187,-98986,-32958,4280,-79129,-6560
4,9344,58964,50584,71128,-55480,24986,15086,-62360,-42977,-49482,-77256,-36895,-
74818,20,3063,-49426,28152,-97329,6086,86035,-88743,35241,44249,19927,-10660,894
04,24179,-26621,-6511,57745,-28750,96340,-97160,-97822,-49979,52307,79462,94273,
-24808,77104,9255,-83057,77655,21361,55956,-9096,48599,-40490,-55107,2689,29608,
20497,66834,-34678,23553,-81400,-66630,-96321,-34499,-12957,-20564,25610,-4322,-
58462,20801,53700,71527,24669,-54534,57879,-3221,33636,3900,97832,-27688,-98715,
5992,24520,-55401,-57613,-69926,57377,-77610,20123,52174,860,60429,-91994,-62403
,-6218,-90610,-37263,-15052,62069,-96465,44254,89892,-3406,19121,-41842,-87783,-
64125,-56120,73904,-22797,-58118,-4866,5356,75318,46119,21276,-19246,-9241,-9742
5,57333,-15802,93149,25689,-5532,95716,39209,-87672,-29470,-16324,-15331,27632,-
39454,56530,-16000,29853,46475,78242,-46602,83192,-73440,-15816,50964,-36601,897
58,38375,-40007,-36675,-94030,67576,46811,-64919,45595,76530,40398,35845,41791,6
7697,-30439,-82944,63115,33447,-36046,-50122,-34789,43003,-78947,-38763,-89210,3
2756,-20389,-31358,-90526,-81607,88741,86643,98422,47389,-75189,13091,95993,-155
01,94260,-25584,-1483,-67261,-70753,25160,89614,-90620,-48542,83889,-12388,-9642
,-37043,-67663,28794,-8801,13621,12241,55379,84290,21692,-95906,-85617,-17341,-6
3767,80183,-4942,-51478,30997,-13658,8838,17452,-82869,-39897,68449,31964,98158,
-49489,62283,-62209,-92792,-59342,55146,-38533,20496,62667,62593,36095,-12470,54
53,-50451,74716,-17902,3302,-16760,-71642,-34819,96459,-72860,21638,47342,-69897
,-40180,44466,76496,84659,13848,-91600,-90887,-63742,-2156,-84981,-99280,94326,-
33854,92029,-50811,98711,-36459,-75555,79110,-88164,-97397,-84217,97457,64387,30
513,-53190,-83215,252,2344,-27177,-92945,-89010,82662,-11670,86069,53417,42702,9
7082,3695,-14530,-46334,17910,77999,28009,-12374,15498,-46941,97088,-35030,95040
,92095,-59469,-24761,46491,67357,-66658,37446,-65130,-50416,99197,30925,27308,54
122,-44719,12582,-99525,-38446,-69050,-22352,94757,-56062,33684,-40199,-46399,96
842,-50881,-22380,-65021,40582,53623,-76034,77018,-97074,-84838,-22953,-74205,79
715,-33920,-35794,-91369,73421,-82492,63680,-14915,-33295,37145,76852,-69442,601
25,-74166,74308,-1900,-30195,-16267,-60781,-27760,5852,38917,25742,-3765,49097,-
63541,98612,-92865,-30248,9612,-8798,53262,95781,-42278,-36529,7252,-27394,-5021
,59178,80934,-48480,-75131,-54439,-19145,-48140,98457,-6601,-51616,-89730,78028,
32083,-48904,16822,-81153,-8832,48720,-80728,-45133,-86647,-4259,-40453,2590,286
13,50523,-4105,-27790,-74579,-17223,63721,33489,-47921,97628,-97691,-14782,-6564
4,18008,-93651,-71266,80990,-76732,-47104,35368,28632,59818,-86269,-89753,34557,
-92230,-5933,-3487,-73557,-13174,-43981,-43630,-55171,30254,-83710,-99583,-13500
,71787,5017,-25117,-78586,86941,-3251,-23867,-36315,75973,86272,-45575,77462,-98
836,-10859,70168,-32971,-38739,-12761,93410,14014,-30706,-77356,-85965,-62316,63
918,-59914,-64088,1591,-10957,38004,15129,-83602,-51791,34381,-89382,-26056,8942
,5465,71458,-73805,-87445,-19921,-80784,69150,-34168,28301,-68955,18041,6059,823
42,9947,39795,44047,-57313,48569,81936,-2863,-80932,32976,-86454,-84207,33033,32
```

867,9104,-16580,-25727,80157,-70169,53741,86522,84651,68480,84018,61932,7332,-61
322,-69663,76370,41206,12326,-34689,17016,82975,-23386,39417,72793,44774,-96259,
3213,79952,29265,-61492,-49337,14162,65886,3342,-41622,-62659,-90402,-24751,8851
1,54739,-21383,-40161,-96610,-24944,-602,-76842,-21856,69964,43994,-15121,-85530
,12718,13170,-13547,69222,62417,-75305,-81446,-38786,-52075,-23110,97681,-82800,
-53178,11474,35857,94197,-58148,-23689,32506,92154,-64536,-73930,-77138,97446,-8
3459,70963,22452,68472,-3728,-25059,-49405,95129,-6167,12808,99918,30113,-12641,
-26665,86362,-33505,50661,26714,33701,89012,-91540,40517,-12716,-57185,-87230,29
914,-59560,13200,-72723,58272,23913,-45586,-96593,-26265,-2141,31087,81399,92511
,-34049,20577,2803,26003,8940,42117,40887,-82715,38269,40969,-50022,72088,21291,
-67280,-16523,90535,18669,94342,-39568,-88080,-99486,-20716,23108,-28037,63342,3
6863,-29420,-44016,75135,73415,16059,-4899,86893,43136,-7041,33483,-67612,25327,
40830,6184,61805,4247,81119,-22854,-26104,-63466,63093,-63685,60369,51023,51644,
-16350,74438,-83514,99083,10079,-58451,-79621,48471,67131,-86940,99093,11855,-22
272,-67683,-44371,9541,18123,37766,-70922,80385,-57513,-76021,-47890,36154,72935
,84387,-92681,-88303,-7810,59902,-90,-64704,-28396,-66403,8860,13343,33882,85680
,7228,28160,-14003,54369,-58893,92606,-63492,-10101,64714,58486,29948,-44679,-22
763,10151,-56695,4031,-18242,-36232,86168,-14263,9883,47124,47271,92761,-24958,-
73263,-79661,-69147,-18874,29546,-92588,-85771,26451,-86650,-43306,-59094,-47492
,-34821,-91763,-47670,33537,22843,67417,-759,92159,63075,94065,-26988,55276,6590
3,30414,-67129,-99508,-83092,-91493,-50426,14349,-83216,-76090,32742,-5306,-9331
0,-60750,-60620,-45484,-21108,-58341,-28048,-52803,69735,78906,81649,32565,-8680
4,-83202,-65688,-1760,89707,93322,-72750,84134,71900,-37720,19450,-78018,22001,-
23604,26276,-21498,65892,-72117,-89834,-23867,55817,-77963,42518,93123,-83916,63
260,-2243,-97108,85442,-36775,17984,-58810,99664,-19082,93075,-69329,87061,79713
,16296,70996,13483,-74582,49900,-27669,-40562,1209,-20572,34660,83193,75579,7344
,64925,88361,60969,3114,44611,-27445,53049,-16085,-92851,-53306,13859,-33532,866
22,-75666,-18159,-98256,51875,-42251,-27977,-18080,23772,38160,41779,9147,94175,
99905,-85755,62535,-88412,-52038,-68171,93255,-44684,-11242,-104,31796,62346,-54
931,-55790,-70032,46221,56541,-91947,90592,93503,4071,20646,4856,-63598,15396,-5
0708,32138,-85164,38528,-89959,53852,57915,-42421,-88916,-75072,67030,-29066,495
42,-71591,61708,-53985,-43051,28483,46991,-83216,80991,-46254,-48716,39356,-8270
,-47763,-34410,874,-1186,-7049,28846,11276,21960,-13304,-11433,-4913,55754,79616
,70423,-27523,64803,49277,14906,-97401,-92390,91075,70736,21971,-3303,55333,-939
96,76538,54603,-75899,98801,46887,35041,48302,-52318,55439,24574,14079,-24889,83
440,14961,34312,-89260,-22293,-81271,-2586,-71059,-10640,-93095,-5453,-70041,665
43,74012,-11662,-52477,-37597,-70919,92971,-17452,-67306,-80418,7225,-89296,2429
6,86547,37154,-10696,74436,-63959,58860,33590,-88925,-97814,-83664,85484,-8385,-
50879,57729,-74728,-87852,-15524,-91120,22062,28134,80917,32026,49707,-54252,-44
319,-35139,13777,44660,85274,25043,58781,-89035,-76274,6364,-63625,72855,43242,-
35033,12820,-27460,77372,-47578,-61162,-70758,-1343,-4159,64935,56024,-2151,4377
0,19758,-30186,-86040,24666,-62332,-67542,73180,-25821,-27826,-45504,-36858,-120
41,20017,-24066,-56625,-52097,-47239,-90694,8959,7712,-14258,-5860,55349,61808,-
4423,-93703,64681,-98641,-25222,46999,-83831,-54714,19997,-68477,66073,51801,-66
491,52061,-52866,79907,-39736,-68331,68937,91464,98892,910,93501,31295,-85873,27
036,-57340,50412,21,-2445,29471,71317,82093,-94823,-54458,-97410,39560,-7628,664
52,39701,54029,37906,46773,58296,60370,-61090,85501,-86874,71443,-72702,-72047,1
4848,34102,77975,-66294,-36576,31349,52493,-70833,-80287,94435,39745,-98291,8452
4,-18942,10236,93448,50846,94023,-6939,47999,14740,30165,81048,84935,-19177,-135
94,32289,62628,-90612,-542,-66627,64255,71199,-83841,-82943,-73885,8623,-67214,-
9474,-35249,62254,-14087,-90969,21515,-83303,94377,-91619,19956,-98810,96727,-91
939,29119,-85473,-82153,-69008,44850,74299,-76459,-86464,8315,-49912,-28665,5905
2,-69708,76024,-92738,50098,18683,-91438,18096,-19335,35659,91826,15779,-73070,6
7873,-12458,-71440,-46721,54856,97212,-81875,35805,36952,68498,81627,-34231,8171
2,27100,-9741,-82612,18766,-36392,2759,41728,69743,26825,48355,-17790,17165,5655
8,3295,-24375,55669,-16109,24079,73414,48990,-11931,-78214,90745,19878,35673,-15
317,-89086,94675,-92513,88410,-93248,-19475,-74041,-19165,32329,-26266,-46828,-1
8747,45328,8990,-78219,-25874,-74801,-44956,-54577,-29756,-99822,-35731,-18348,-

68915,-83518,-53451,95471,-2954,-13706,-8763,-21642,-37210,16814,-60070,-42743,27697,-36333,-42362,11576,85742,-82536,68767,-56103,-63012,71396,-78464,-68101,-15917,-11113,-3596,77626,-60191,-30585,-73584,6214,-84303,18403,23618,-15619,-89755,-59515,-59103,-74308,-63725,-29364,-52376,-96130,70894,-12609,50845,-2314,42264,-70825,64481,55752,4460,-68603,-88701,4713,-50441,-51333,-77907,97412,-66616,-49430,60489,-85262,-97621,-18980,44727,-69321,-57730,66287,-92566,-64427,-14270,11515,-92612,-87645,61557,24197,-81923,-39831,-10301,-23640,-76219,-68025,92761,-76493,68554,-77734,-95620,-11753,-51700,98234,-68544,-61838,29467,46603,-18221,-35441,74537,40327,-58293,75755,-57301,-7532,-94163,18179,-14388,-22258,-46417,-48285,18242,-77551,82620,250,-20060,-79568,-77259,82052,-98897,-75464,48773,-79040,-11293,45941,-67876,-69204,-46477,-46107,792,60546,-34573,-12879,-94562,20356,-48004,-62429,96242,40594,2099,99494,25724,-39394,-2388,-18563,-56510,-83570,-29214,3015,74454,74197,76678,-46597,60630,-76093,37578,-82045,-24077,62082,-87787,-74936,58687,12200,-98952,70155,-77370,21710,-84625,-60556,-84128,925,65474,-15741,-94619,88377,89334,44749,22002,-45750,-93081,-14600,-83447,46691,85040,-66447,-80085,56308,44310,24979,-29694,57991,4675,-71273,-44508,13615,-54710,23552,-78253,-34637,50497,68706,81543,-88408,-21405,6001,-33834,-21570,-46692,-25344,20310,71258,-97680,11721,59977,59247,-48949,98955,-50276,-80844,-27935,-76102,55858,-33492,40680,66691,-33188,8284,64893,-7528,6019,-85523,8434,-64366,-56663,26862,30008,-7611,-12179,-70076,21426,-11261,-36864,-61937,-59677,929,-21052,3848,-20888,-16065,98995,-32293,-86121,-54564,77831,68602,74977,31658,40699,29755,98424,80358,-69337,26339,13213,-46016,-18331,64713,-46883,-58451,-70024,-92393,-4088,70628,-51185,71164,-75791,-1636,-29102,-16929,-87650,-84589,-24229,-42137,-15653,94825,13042,88499,-47100,-90358,-7180,29754,-65727,-42659,-85560,-9037,-52459,20997,-47425,17318,21122,20472,-23037,65216,-63625,-7877,-91907,24100,-72516,22903,-85247,-8938,73878,54953,87480,-31466,-99524,35369,-78376,89984,-15982,94045,-7269,23319,-80456,-37653,-76756,2909,81936,54958,-12393,60560,-84664,-82413,66941,-26573,-97532,64460,18593,-85789,-38820,-92575,-43663,-89435,83272,-50585,13616,-71541,-53156,727,-27644,16538,34049,57745,34348,35009,16634,-18791,23271,-63844,95817,21781,16590,59669,15966,-6864,48050,-36143,97427,-59390,96931,78939,-1958,50777,43338,-51149,39235,-27054,-43492,67457,-83616,37179,10390,85818,2391,73635,87579,-49127,-81264,-79023,-81590,53554,-74972,-83940,-13726,-39095,29174,78072,76104,47778,25797,-29515,-6493,-92793,22481,-36197,-65560,42342,15750,97556,99634,-56048,-35688,13501,63969,-74291,50911,39225,93702,-3490,-59461,-30105,-46761,-80113,92906,-68487,50742,36152,-90240,-83631,24597,-50566,-15477,18470,77038,40223,-80364,-98676,70957,-63647,99537,13041,31679,86631,37633,-16866,13686,-71565,21652,-46053,-80578,-61382,68487,-6417,4656,20811,67013,-30868,-11219,46,74944,14627,56965,42275,-52480,52162,-84883,-52579,-90331,92792,42184,-73422,-58440,65308,-25069,5475,-57996,59557,-17561,2826,-56939,14996,-94855,-53707,99159,43645,-67719,-1331,21412,41704,31612,32622,1919,-69333,-69828,22422,-78842,57896,-17363,27979,-76897,35008,46482,-75289,65799,20057,7170,41326,-76069,90840,-81253,-50749,3649,-42315,45238,-33924,62101,96906,58884,-7617,-28689,-66578,62458,50876,-57553,6739,41014,-64040,-34916,37940,13048,-97478,-11318,-89440,-31933,-40357,-59737,-76718,-14104,-31774,28001,4103,41702,-25120,-31654,63085,-3642,84870,-83896,-76422,-61520,12900,88678,85547,33132,-88627,52820,63915,-27472,78867,-51439,33005,-23447,-3271,-39308,39726,-74260,-31874,-36893,93656,910,-98362,60450,-88048,99308,13947,83996,-90415,-35117,70858,-55332,-31721,97528,82982,-86218,6822,25227,36946,97077,-4257,-41526,56795,89870,75860,-70802,21779,14184,-16511,-89156,-31422,71470,69600,-78498,74079,-19410,40311,28501,26397,-67574,-32518,68510,38615,19355,-6088,-97159,-29255,-92523,3023,-42536,-88681,64255,41206,44119,52208,39522,-52108,91276,-70514,83436,63289,-79741,9623,99559,12642,85950,83735,-21156,-67208,98088,-7341,-27763,-30048,-44099,-14866,-45504,-91704,19369,13700,10481,-49344,-85686,33994,19672,36028,60842,66564,-24919,33950,-93616,-47430,-35391,-28279,56806,74690,39284,-96683,-7642,-75232,37657,-14531,-86870,-9274,-26173,98640,88652,64257,46457,37814,-19370,9337,-22556,-41525,39105,-28719,51611,-93252,98044,-90996,21710,-47605,-64259,-32727,53611,-31918,-3555,33316,-66472,21274,-37731,-2919,15016,48779,-88868,1897,41728,46344,-89667,37848,68092,-44011,85354,-43776,38739,-31423,-66330,65167,-22016,59405,34328,-60042,87660,-67698,-59174,-

1408,-46809,-43485,-88807,-60489,13974,22319,55836,-62995,-37375,-4185,32687,-36
551,-75237,58280,26942,-73756,71756,78775,-40573,14367,-71622,-77338,24112,23414
,-7679,-51721,87492,85066,-21612,57045,10673,-96836,52461,-62218,-9310,65862,-22
748,89906,-96987,-98698,26956,-43428,46141,47456,28095,55952,67323,-36455,-60202
,-43302,-82932,42020,77036,10142,60406,70331,63836,58850,-66752,52109,21395,-102
38,-98647,-41962,27778,69060,98535,-28680,-52263,-56679,66103,-42426,27203,80021
,10153,58678,36398,63112,34911,20515,62082,-15659,-40785,27054,43767,-20289,6583
8,-6954,-60228,-72226,52236,-35464,25209,-15462,-79617,-41668,-84083,62404,-6906
2,18913,46545,20757,13805,24717,-18461,-47009,-25779,68834,64824,34473,39576,315
70,14861,-15114,-41233,95509,68232,67846,84902,-83060,17642,-18422,73688,77671,-
26930,64484,-99637,73875,6428,21034,-73471,19664,-68031,15922,-27028,48137,54955
,-82793,-41144,-10218,-24921,-28299,-2288,68518,-54452,15686,-41814,66165,-72207
,-61986,80020,50544,-99500,16244,78998,40989,14525,-56061,-24692,-94790,21111,37
296,-90794,72100,70550,-31757,17708,-74290,61910,78039,-78629,-25033,73172,-9195
3,10052,64502,99585,-1741,90324,-73723,68942,28149,30218,24422,16659,10710,-6259
4,94249,96588,46192,34251,73500,-65995,-81168,41412,-98724,-63710,-54696,-52407,
19746,45869,27821,-94866,-76705,-13417,-61995,-71560,43450,67384,-8838,-80293,-2
8937,23330,-89694,-40586,46918,80429,-5475,78013,25309,-34162,37236,-77577,86744
,26281,-29033,-91813,35347,13033,-13631,-24459,3325,-71078,-75359,81311,19700,47
678,-74680,-84113,45192,35502,37675,19553,76522,-51098,-18211,89717,4508,-82946,
27749,85995,89912,-53678,-64727,-14778,32075,-63412,-40524,86440,-2707,-36821,63
850,-30883,67294,-99468,-23708,34932,34386,98899,29239,-23385,5897,54882,98660,4
9098,70275,17718,88533,52161,63340,50061,-89457,19491,-99156,24873,-17008,64610,
-55543,50495,17056,-10400,-56678,-29073,-42960,-76418,98562,-88104,-96255,10159,
-90724,54011,12052,45871,-90933,-69420,67039,37202,78051,-52197,-40278,-58425,65
414,-23394,-1415,6912,-53447,7352,17307,-78147,63727,98905,55412,-57658,-32884,-
44878,22755,39730,3638,35111,39777,74193,38736,-11829,-61188,-92757,55946,-71232
,-63032,-83947,39147,-96684,-99233,25131,-32197,24406,-55428,-61941,25874,-69453
,64483,-19644,-68441,12783,87338,-48676,66451,-447,-61590,50932,-11270,29035,656
98,-63544,10029,80499,-9461,86368,91365,-81810,-71914,-52056,-13782,44240,-30093
,-2437,24007,67581,-17365,-69164,-8420,-69289,-29370,48010,90439,13141,69243,506
68,39328,61731,78266,-81313,17921,-38196,55261,9948,-24970,75712,-72106,28696,74
61,31621,61047,51476,56512,11839,-96916,-82739,28924,-99927,58449,37280,69357,11
219,-32119,-62050,-48745,-83486,-52376,42668,82659,68882,38773,46269,-96005,9763
0,25009,-2951,-67811,99801,81587,-79793,-18547,-83086,69512,33127,-92145,-88497,
47703,59527,1909,88785,-88882,69188,-46131,-5589,-15086,36255,-53238,-33009,8266
4,53901,35939,-42946,-25571,33298,69291,53199,74746,-40127,-39050,91033,51717,-9
8048,87240,36172,65453,-94425,-63694,-30027,59004,88660,3649,-20267,-52565,-6732
1,34037,4320,91515,-56753,60115,27134,68617,-61395,-26503,-98929,-8849,-63318,10
709,-16151,61905,-95785,5262,23670,-25277,90206,-19391,45735,37208,-31992,-92450
,18516,-90452,-58870,-58602,93383,14333,17994,82411,-54126,-32576,35440,-60526,-
78764,-25069,-9022,-394,92186,-38057,55328,-61569,67780,77169,19546,-92664,-9494
8,44484,-13439,83529,27518,-48333,72998,38342,-90553,-98578,-76906,81515,-16464,
78439,92529,35225,-39968,-10130,-7845,-32245,-74955,-74996,67731,-13897,-82493,3
3407,93619,59560,-24404,-57553,19486,-45341,34098,-24978,-33612,79058,71847,7671
3,-95422,6421,-96075,-59130,-28976,-16922,-62203,69970,68331,21874,40551,89650,5
1908,58181,66480,-68177,34323,-3046,-49656,-59758,43564,-10960,-30796,15473,-202
16,46085,-85355,41515,-30669,-87498,57711,56067,63199,-83805,62042,91213,-14606,
4394,-562,74913,10406,96810,-61595,32564,31640,-9732,42058,98052,-7908,-72330,15
58,-80301,34878,32900,3939,-8824,88316,20937,21566,-3218,-66080,-31620,86859,542
89,90476,-42889,-15016,-18838,75456,30159,-67101,42328,-92703,85850,-5475,23470,
-80806,68206,17764,88235,46421,-41578,74005,-81142,80545,20868,-1560,64017,83784
,68863,-97516,-13016,-72223,79630,-55692,82255,88467,28007,-34686,-69049,-41677,
88535,-8217,68060,-51280,28971,49088,49235,26905,-81117,-44888,40623,74337,-2466
2,97476,79542,-72082,-35093,98175,-61761,-68169,59697,-62542,-72965,59883,-64026
,-37656,-92392,-12113,-73495,98258,68379,-21545,64607,-70957,-92254,-97460,-6343
6,-8853,-19357,-51965,-76582,12687,-49712,45413,-60043,33496,31539,-57347,41837,
67280,-68813,52088,-13155,-86430,-15239,-45030,96041,18749,-23992,46048,35243,-7

```
9450,85425,-58524,88781,-39454,53073,-48864,-82289,39086,82540,-11555,25014,-543
1,-39585,-89526,2705,31953,-81611,36985,-56022,68684,-27101,11422,64655,-26965,-
63081,-13840,-91003,-78147,-8966,41488,1988,99021,-61575,-47060,65260,-23844,-21
781,-91865,-19607,44808,2890,63692,-88663,-58272,15970,-65195,-45416,-48444,-782
26,-65332,-24568,42833,-1806,-71595,80002,-52250,30952,48452,-90106,31015,-22073
,62339,63318,78391,28699,77900,-4026,-76870,-45943,33665,9174,-84360,-22684,-168
32,-67949,-38077,-38987,-32847,51443,-53580,-13505,9344,-92337,26585,70458,-5276
4,-67471,-68411,-1119,-2072,-93476,67981,40887,-89304,-12235,41488,1454,5355,-34
855,-72080,24514,-58305,3340,34331,8731,77451,-64983,-57876,82874,62481,-32754,-
39902,22451,-79095,-23904,78409,-7418,77916};
    vector< vector<int> > r = s.threeSum(v);
    vector<int> vec_tmp;
    for(vector< vector<int>>::iterator iter=r.begin();iter!=r.end();++iter){
        vec_tmp = *iter;
        for(vector<int>::iterator it=vec_tmp.begin();it!=vec_tmp.end();++it){
            cout<<*it<<" ";
        }
        cout<<endl;
    }
    return 0;
}
```

这个长的测试用例，也用了30秒的时间。。。