Code Jam 2008 - World Finals

Analysis: Juice

In this problem, we need to find the best (A*, B*, C*) such that there is a maximum number of (A, B, C) triplets in the input satisfying:

> $A \le A^*$, $B \le B^*$, $C \le C^*$, and $A^* + B^* + C^* \le 10000$.

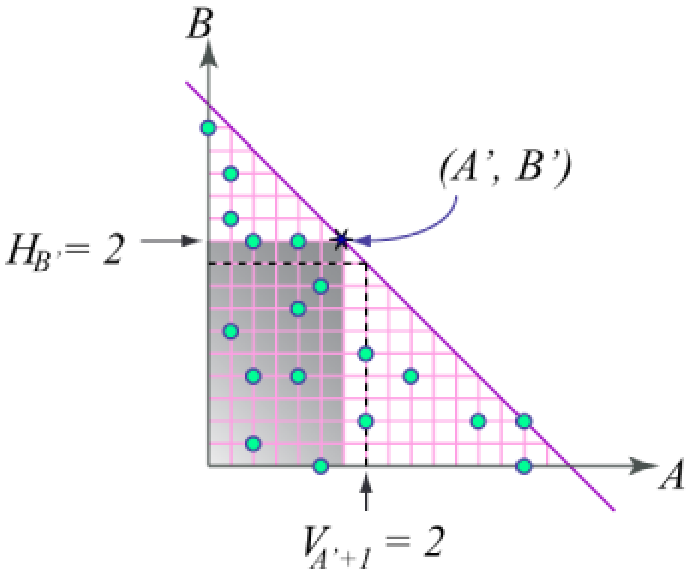
It is easy to see that we can just consider integer A*, B*, C*'s. In fact C* can be one of the C's from the input, so are A* and B* -- otherwise we can just decrease it until it hits some C for a satisfied customer.

So we have at most 5000 possible candidate values for C* (or 10000, if you don't want to use the above observation). We try each of them. And the problem is nicely visualized in 2-dimensional grid.

For a fixed C*, we know $A^* + B^* \le 10000 - C^*$. We filter out all the inputs that have

> $C \le C^*$ or $A + B \ge 10000 - C^*$,

and view the remaining as points in the 2-d plane, with their A, B as the coordinates. For the best solution (A*, B*), we can just try all the integer points (10001 - C* of them) on the line $A^* + B^* = 10000 - C^*$. For each point, we need to know *quickly* how many input points are dominated by that point, i.e., lying in the axis-parallel rectangle between the origin and that point.



The last step must be computed fast enough to meet the time limit of the competition. Assume we move the point (A*, B*) from the top-left to bottom-right. In a certain step, we are at (A', B'), with Q points dominated by it. In the next step, we are at (A' + 1, B' - 1), the number of points dominated by the new point can be computed as

> $Q' = Q - H_{B'} + V_{A' + 1}$,

where V_A is the counter of points on the A-th vertical line, and H_B is the number of points on the B-th horizontal line. Q' can be computed in constant time if we pre-compute the counters.

Solution from the judges:

```
int T, n, ans;
int A[5000], B[5000], C[5000], H[10001], V[10001];

int main() {
  cin>>T;
  for (int t=1; t<=T; t++) {
    cin>>n;
    for(int i=0; i<n; ++i) cin>>A[i]>>B[i]>>C[i];

    int ans = 0;
    for (int CC=0; CC<=10000; ++CC) {
      memset(H, 0, sizeof(H));
      memset(V, 0, sizeof(V));
      for (int i=0; i<n; ++i)
        if (C[i]<=CC && A[i]+B[i]+CC<=10000)
        { V[A[i]]++; H[B[i]]++; }
      int Q = 0;
      for (int AA=-1; AA<10000-CC; ++AA) {
        Q = Q + V[AA+1] - H[10000-CC-AA];
        ans >?= Q;
      }
    }

    cout<<"Case #"<<t<<": "<<ans<<endl;
  }
  return 0;
}
```