```cpp
#include<stdio.h>
#include<string.h>
#include<math.h>
#include<utility>
#include<vector>
#include<queue>
using namespace std;

#define MAX 512
#define INF 112345678

typedef struct{ int x, y; }ponto;
typedef pair<double, int> di;
typedef vector<di> vdi;

int n, seen[MAX];
double dist[MAX];

int disteu(ponto a, ponto b) {
  int r1 = a.x - b.x, r2 = a.y - b.y;
  return r1 * r1 + r2 * r2;
}

double prim(int s, vector<vdi> &LG) {
  int u, i, v;
  double ret = 0, c;
  priority_queue< di, vector<di>, greater<di> > Q;
  memset(seen, 0, sizeof(seen));
  for (u = 0; u < n; u++) dist[u] = INF;
  dist[s] = 0; Q.push(di(0, s));
  while (!Q.empty()) {
    u = Q.top().second; Q.pop();
    if (seen[u]) continue;
    seen[u] = 1; ret += dist[u];
    for (i = 0; i < (int)LG[u].size(); i++) {
      c = LG[u][i].first; v = LG[u][i].second;
      if (dist[v] > c) {
        dist[v] = c; Q.push(di(c, v));
      }}}
  return ret;
}

int main(void) {
  int c, i, j;
  ponto t[MAX];
  scanf("%d", &c);
  while (c--) {
    scanf("%d", &n);
    vector<vdi> LG(n);
    for (i = 0; i < n; i++) scanf("%d %d", &t[i].x, &t[i].y);
    for (i = 0; i < n; i++)
      for (j = i+1; j < n; j++) {
    LG[i].push_back(di(sqrt(disteu(t[i], t[j])), j));
    LG[j].push_back(di(sqrt(disteu(t[i], t[j])), i));
      }
    printf("%.2lf\n", prim(0, LG) / 100);
  }
  return 0;
}
```