#include <iostream>  
#include <vector>  
#include <cmath>  
#include <string>  
using namespace std;  
  
struct Linha {  
 string nome;  
 double km\_rota;  
 int num\_paradas;  
 int num\_viagens;  
  
 double cobertura;  
 double capacidade;  
 double frequencia;  
  
 double ux\_cob, ux\_cap, ux\_freq, otc;  
 string nivel;  
};  
  
double media(const vector<double>& valores) {  
 double soma = 0;  
 for (double v : valores) soma += v;  
 return soma / valores.size();  
}  
  
double desvio\_padrao(const vector<double>& valores, double m) {  
 double soma = 0;  
 for (double v : valores) soma += pow(v - m, 2);  
 return sqrt(soma / valores.size());  
}  
  
string classificar\_otc(double otc) {  
 if (otc >= 1.5) return "A";  
 else if (otc >= 1.0) return "B";  
 else if (otc >= 0.5) return "C";  
 else if (otc >= 0.0) return "D";  
 else if (otc >= -1.0) return "E";  
 else return "F";  
}  
  
int main() {  
 double populacao = 110150;  
 double area = 2540;  
 int lugares = 80;  
  
 vector<Linha> linhas = {  
 {"Cajueiro", 13.035, 82, 8},  
 {"Campus", 5.137, 32, 101},  
 {"Ganchária", 14.054, 92, 35},  
 {"Gapara", 11.71, 84, 38},  
 {"Pianco", 11.922, 72, 52},  
 {"Vila Maranhão", 13.021, 74, 9}  
 };  
  
 vector<double> coberturas, capacidades, frequencias;  
  
 for (auto& linha : linhas) {  
 linha.cobertura = linha.num\_paradas / area;  
 linha.capacidade = (linha.num\_viagens \* lugares) / populacao;  
 linha.frequencia = linha.num\_viagens;  
 coberturas.push\_back(linha.cobertura);  
 capacidades.push\_back(linha.capacidade);  
 frequencias.push\_back(linha.frequencia);  
 }  
  
 double m\_cob = media(coberturas);  
 double dp\_cob = desvio\_padrao(coberturas, m\_cob);  
 double m\_cap = media(capacidades);  
 double dp\_cap = desvio\_padrao(capacidades, m\_cap);  
 double m\_freq = media(frequencias);  
 double dp\_freq = desvio\_padrao(frequencias, m\_freq);  
  
 for (auto& linha : linhas) {  
 linha.ux\_cob = (linha.cobertura - m\_cob) / dp\_cob;  
 linha.ux\_cap = (linha.capacidade - m\_cap) / dp\_cap;  
 linha.ux\_freq = (linha.frequencia - m\_freq) / dp\_freq;  
 linha.otc = linha.ux\_cob + linha.ux\_cap + linha.ux\_freq;  
 linha.nivel = classificar\_otc(linha.otc);  
 }  
  
 cout << "Linha\tCobertura\tCapacidade\tFrequencia\tOTC\tNivel\n";  
 for (auto& linha : linhas) {  
 cout << linha.nome << "\t"  
 << linha.cobertura << "\t"  
 << linha.capacidade << "\t"  
 << linha.frequencia << "\t"  
 << linha.otc << "\t"  
 << linha.nivel << "\n";  
 }  
  
 return 0;  
}