



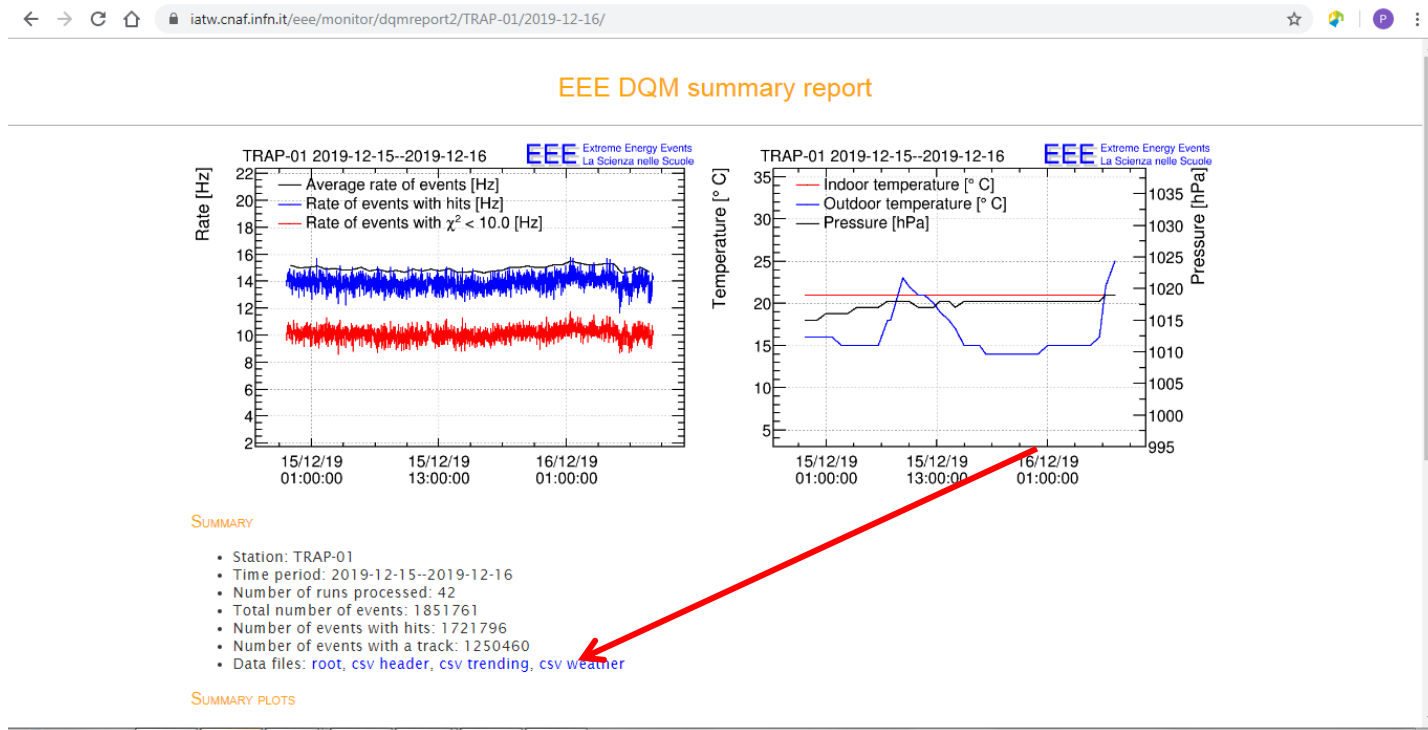
ROOT @ EEE

Lezione 2

Realizzazione di grafici per punti (TGraph)

Lezione precedente: lettura e scrittura da file testo

https://agenda.centrofermi.it/event/162/contributions/1367/attachments/690/1026/ROOT_lesson1.pdf



Lezione precedente: lettura e scrittura da file testo

```
{  
    // inizio della macro  
  
    ifstream filedati("D:\\EEE\\ROOT lessons\\weather_trending.csv");  
    if(filedati.fail())  
    {  
        cout << "Il file non esiste, verifica il percorso!" << endl;  
        break;  
    }  
  
    string titolo1, titolo2, titolo3, titolo4, titolo5;  
    double secondi, Tin, Tout, pressione, run;  
  
    filedati >> titolo1 >> titolo2 >> titolo3 >> titolo4 >> titolo5;  
  
    double Tin_media = 0;  
    int nrighe = 0;  
  
    while(filedati >> secondi >> Tin >> Tout >> pressione >> run)  
    {  
        Tin_media = Tin_media + Tin;  
        nrighe++;  
    }  
  
    Tin_media = Tin_media / (double) nrighe;  
    cout << Tin_media << endl;  
    ofstream fileout("D:\\EEE\\ROOT lessons\\out.txt");  
    fileout << Tin_media ;  
  
}
```

Lezione precedente: lettura e scrittura da file testo

```
{
    // inizio della macro
    ifstream filedati("D:\\EEE\\ROOT lessons\\weather_trending.csv");
    if(filedati.fail())
    {
        cout << "Il file non esiste, verifica il percorso!" << endl;
        break;
    }
    string titolo1, titolo2, titolo3, titolo4, titolo5;
    double secondi, Tin, Tout, pressione, run;

    filedati >> titolo1 >> titolo2 >> titolo3 >> titolo4 >> titolo5;

    double Tin_media = 0;
    int nrighe = 0;

    while(filedati >> secondi >> Tin >> Tout >> pressione >> run)
    {
        Tin_media = Tin_media + Tin;
        nrighe++;
    }
    Tin_media = Tin_media / (double) nrighe;
    cout << Tin_media << endl;
    ofstream fileout("D:\\EEE\\ROOT lessons\\out.txt");
    fileout << Tin_media ;
}
```

Percorso file dati

Lezione precedente: lettura e scrittura da file testo

```
{
    // inizio della macro

    ifstream filedati("D:\\EEE\\ROOT lessons\\weather_trending.csv");
    if(filedati.fail())
    {
        cout << "Il file non esiste, verifica il percorso!" << endl;
        break;
    }

    string titolo1, titolo2, titolo3, titolo4, titolo5;
    double secondi, Tin, Tout, pressione, run;

    filedati >> titolo1 >> titolo2 >> titolo3 >> titolo4 >> titolo5;

    double Tin_media = 0;
    int nrighe = 0;

    while(filedati >> secondi >> Tin >> Tout >> pressione >> run)
    {
        Tin_media = Tin_media + Tin;
        nrighe++;
    }

    Tin_media = Tin_media / (double) nrighe;
    cout << Tin_media << endl;
    ofstream fileout("D:\\EEE\\ROOT lessons\\out.txt");
    fileout << Tin_media ;

}
```

Percorso file dati
**Verifica
esistenza
del file**

Lezione precedente: lettura e scrittura da file testo

```
{
    // inizio della macro

    ifstream filedati("D:\\EEE\\ROOT lessons\\weather_trending.csv");
    if(filedati.fail())
    {
        cout << "Il file non esiste, verifica il percorso!" << endl;
        break;
    }

    string titolo1, titolo2, titolo3, titolo4, titolo5;
    double secondi, Tin, Tout, pressione, run;

    filedati >> titolo1 >> titolo2 >> titolo3 >> titolo4 >> titolo5;

    double Tin_media = 0;
    int nrighe = 0;

    while(filedati >> secondi >> Tin >> Tout >> pressione >> run)
    {
        Tin_media = Tin_media + Tin;
        nrighe++;
    }
    Tin_media = Tin_media / (double) nrighe;
    cout << Tin_media << endl;
    ofstream fileout("D:\\EEE\\ROOT lessons\\out.txt");
    fileout << Tin_media ;

}
```

Percorso file dati

**Verifica
esistenza
del file**

**Dichiarazione variabili utilizzate
per dati in lettura**

Lezione precedente: lettura e scrittura da file testo

```
{
    // inizio della macro

    ifstream filedati("D:\\EEE\\ROOT lessons\\weather_trending.csv");
    if(filedati.fail())
    {
        cout << "Il file non esiste, verifica il percorso!" << endl;
        break;
    }

    string titolo1, titolo2, titolo3, titolo4, titolo5;
    double secondi, Tin, Tout, pressione, run;

    filedati >> titolo1 >> titolo2 >> titolo3 >> titolo4 >> titolo5;

    double Tin_media = 0;
    int nrighe = 0;

    while(filedati >> secondi >> Tin >> Tout >> pressione >> run)
    {
        Tin_media = Tin_media + Tin;
        nrighe++;
    }
    Tin_media = Tin_media / (double) nrighe;
    cout << Tin_media << endl;
    ofstream fileout("D:\\EEE\\ROOT lessons\\out.txt");
    fileout << Tin_media ;

}
```

Percorso file dati

**Verifica
esistenza
del file**

**Dichiarazione variabili utilizzate
per dati in lettura**

Lettura prima riga

Lezione precedente: lettura e scrittura da file testo

```
{
    // inizio della macro

    ifstream filedati("D:\\EEE\\ROOT lessons\\weather_trending.csv");
    if(filedati.fail())
    {
        cout << "Il file non esiste, verifica il percorso!" << endl;
        break;
    }

    string titolo1, titolo2, titolo3, titolo4, titolo5;
    double secondi, Tin, Tout, pressione, run;

    filedati >> titolo1 >> titolo2 >> titolo3 >> titolo4 >> titolo5;

    double Tin_media = 0;
    int nrighe = 0;

    while(filedati >> secondi >> Tin >> Tout >> pressione >> run)
    {
        Tin_media = Tin_media + Tin;
        nrighe++;
    }
    Tin_media = Tin_media / (double) nrighe;
    cout << Tin_media << endl;
    ofstream fileout("D:\\EEE\\ROOT lessons\\out.txt");
    fileout << Tin_media ;
}
```

Percorso file dati

**Verifica
esistenza
del file**

**Dichiarazione variabili utilizzate
per dati in lettura**

Lettura prima riga

Dichiarazione variabili utilizzate per calcolo media

Lezione precedente: lettura e scrittura da file testo

```
{
    // inizio della macro

    ifstream filedati("D:\\EEE\\ROOT lessons\\weather_trending.csv");
    if(filedati.fail())
    {
        cout << "Il file non esiste, verifica il percorso!" << endl;
        break;
    }

    string titolo1, titolo2, titolo3, titolo4, titolo5;
    double secondi, Tin, Tout, pressione, run;

    filedati >> titolo1 >> titolo2 >> titolo3 >> titolo4 >> titolo5;

    double Tin_media = 0;
    int nrighe = 0;

    while(filedati >> secondi >> Tin >> Tout >> pressione >> run)
    {
        Tin_media = Tin_media + Tin;
        nrighe++;
    }
    Tin_media = Tin_media / (double) nrighe;
    cout << Tin_media << endl;
    ofstream fileout("D:\\EEE\\ROOT lessons\\out.txt");
    fileout << Tin_media ;

}
```

Percorso file dati

**Verifica
esistenza
del file**

**Dichiarazione variabili utilizzate
per dati in lettura**

Lettura prima riga

Dichiarazione variabili utilizzate per calcolo media

**Ciclo di lettura, calcolo
della media**

Lezione precedente: lettura e scrittura da file testo

```
{
    // inizio della macro

    ifstream filedati("D:\\EEE\\ROOT lessons\\weather_trending.csv");
    if(filedati.fail())
    {
        cout << "Il file non esiste, verifica il percorso!" << endl;
        break;
    }

    string titolo1, titolo2, titolo3, titolo4, titolo5;
    double secondi, Tin, Tout, pressione, run;

    filedati >> titolo1 >> titolo2 >> titolo3 >> titolo4 >> titolo5;

    double Tin_media = 0;
    int nrighe = 0;

    while(filedati >> secondi >> Tin >> Tout >> pressione >> run)
    {
        Tin_media = Tin_media + Tin;
        nrighe++;
    }
    Tin_media = Tin_media / (double) nrighe;
    cout << Tin_media << endl;
    ofstream fileout("D:\\EEE\\ROOT lessons\\out.txt");
    fileout << Tin_media ;
}
```

Percorso file dati

Verifica esistenza del file

Dichiarazione variabili utilizzate per dati in lettura

Lettura prima riga

Dichiarazione variabili utilizzate per calcolo media

Ciclo di lettura, calcolo della media

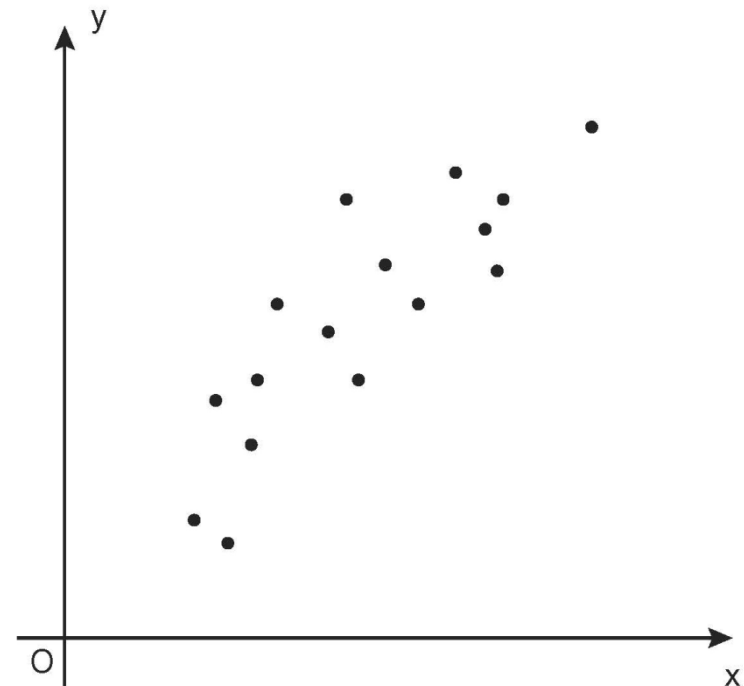
Stampa output su terminale e su file testo

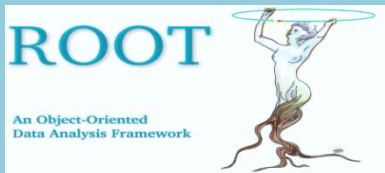
OBIETTIVO: realizzare un grafico per punti

Esempio: riportare l'andamento di una grandezza (come la temperatura) in funzione del tempo

INFO NECESSARIE

- Coordinate punti (x_i, y_i)
 - due array contenenti rispettivamente le coordinate x_i e y_i
- Numero di punti n
- Eventuali errori da associare ai punti
 - due array contenenti rispettivamente le gli errori sulle coordinate x_i e y_i





In ROOT per realizzare un grafico a punti si utilizzano le classi
TGraph o TGraphErrors

Esistono diversi metodi costruttori:

TGraph() → TGraph vuoto (da riempire con altre istruzioni)

TGraph(n) → TGraph di n punti ma senza coordinate (x, y) dei punti (da riempire con altre istruzioni)

TGraph(x, y) → TGraph contenente le coordinate (x, y) dei punti

TGraph(n, x, y) → TGraph contenente le coordinate (x, y) di n punti

...

(vedi documentazione <https://root.cern/root/html534/TGraph.html>)

Nel nostro caso definiamo un TGraph vuoto e assegniamo le coordinate dei punti all'interno del ciclo di lettura dei dati

```

{
    // inizio della macro

    ifstream filedati("D:\\EEE\\ROOT lessons\\weather_trending.csv");
    if(filedati.fail())
    {
        cout << "Il file non esiste, verifica il percorso!" << endl;
        break;
    }
    string titolo1, titolo2, titolo3, titolo4, titolo5;
    double secondi, Tin, Tout, pressione, run;
    TGraph *g = new TGraph();

    filedati >> titolo1 >> titolo2 >> titolo3 >> titolo4 >> titolo5;

    double Tin_media = 0;
    int nrighe = 0;

    while(filedati >> secondi >> Tin >> Tout >> pressione >> run)
    {
        Tin_media = Tin_media + Tin;
        g->SetPoint(nrighe, secondi, Tout);
        nrighe++;
    }
    Tin_media = Tin_media / (double) nrighe;
    cout << Tin_media << endl;
    ofstream fileout("D:\\EEE\\ROOT lessons\\out.txt");
    fileout << Tin_media ;

    TCanvas *c = new TCanvas();
    g->Draw("AP*");
    g->GetXaxis()->SetTimeDisplay(1);
    TDateTime da(2007,01,01,00,00,00);
    gStyle->SetTimeOffset(da.Convert());
    g->GetXaxis()->SetTimeFormat("#splitline{%d\\/%m\\/%y}{%H:%M}");
    g->GetXaxis()->SetNdivisions(10,5,0,kTRUE);
    g->GetXaxis()->SetLabelOffset(0.030);
}

```

```

{
    // inizio della macro

    ifstream filedati("D:\\EEE\\ROOT lessons\\weather_trending.csv");
    if(filedati.fail())
    {
        cout << "Il file non esiste, verifica il percorso!" << endl;
        break;
    }
    string titolo1, titolo2, titolo3, titolo4, titolo5;
    double secondi, Tin, Tout, pressione, run;
    TGraph *g = new TGraph(); Dichiarazione TGraph (vuoto)

    filedati >> titolo1 >> titolo2 >> titolo3 >> titolo4 >> titolo5;

    double Tin_media = 0;
    int nrighe = 0;

    while(filedati >> secondi >> Tin >> Tout >> pressione >> run)
    {
        Tin_media = Tin_media + Tin;
        g->SetPoint(nrighe, secondi, Tout);
        nrighe++;
    }
    Tin_media = Tin_media / (double) nrighe;
    cout << Tin_media << endl;
    ofstream fileout("D:\\EEE\\ROOT lessons\\out.txt");
    fileout << Tin_media ;

    TCanvas *c = new TCanvas();
    g->Draw("AP*");
    g->GetXaxis()->SetTimeDisplay(1);
    TDateTime da(2007,01,01,00,00,00);
    gStyle->SetTimeOffset(da.Convert());
    g->GetXaxis()->SetTimeFormat("#splitline{%d\\/%m\\/%y}{%H:%M}");
    g->GetXaxis()->SetNdivisions(10,5,0,kTRUE);
    g->GetXaxis()->SetLabelOffset(0.030);
}

```

```

{
    // inizio della macro

    ifstream filedati("D:\\EEE\\ROOT lessons\\weather_trending.csv");
    if(filedati.fail())
    {
        cout << "Il file non esiste, verifica il percorso!" << endl;
        break;
    }
    string titolo1, titolo2, titolo3, titolo4, titolo5;
    double secondi, Tin, Tout, pressione, run;
    TGraph *g = new TGraph(); Dichiarazione TGraph (vuoto)

    filedati >> titolo1 >> titolo2 >> titolo3 >> titolo4 >> titolo5;

    double Tin_media = 0;
    int nrighe = 0;

    while(filedati >> secondi >> Tin >> Tout >> pressione >> run)
    {
        Tin_media = Tin_media + Tin;
        g->SetPoint(nrighe, secondi, Tout); Assegnazione coordinate dei punti
        nrighe++;
    }
    Tin_media = Tin_media / (double) nrighe;
    cout << Tin_media << endl;
    ofstream fileout("D:\\EEE\\ROOT lessons\\out.txt");
    fileout << Tin_media ;

    TCanvas *c = new TCanvas();
    g->Draw("AP*");
    g->GetXaxis()->SetTimeDisplay(1);
    TDateTime da(2007,01,01,00,00,00);
    gStyle->SetTimeOffset(da.Convert());
    g->GetXaxis()->SetTimeFormat("#splitline{%d\\/%m\\/%y}{%H:%M}");
    g->GetXaxis()->SetNdivisions(10,5,0,kTRUE);
    g->GetXaxis()->SetLabelOffset(0.030);
}

```



```

{
    // inizio della macro

    ifstream filedati("D:\\EEE\\ROOT lessons\\weather_trending.csv");
    if(filedati.fail())
    {
        cout << "Il file non esiste, verifica il percorso!" << endl;
        break;
    }
    string titolo1, titolo2, titolo3, titolo4, titolo5;
    double secondi, Tin, Tout, pressione, run;
    TGraph *g = new TGraph(); Dichiarazione TGraph (vuoto)

    filedati >> titolo1 >> titolo2 >> titolo3 >> titolo4 >> titolo5;

    double Tin_media = 0;
    int nrighe = 0;

    while(filedati >> secondi >> Tin >> Tout >> pressione >> run)
    {
        Tin_media = Tin_media + Tin;
        g->SetPoint(nrighe, secondi, Tout); Assegnazione coordinate dei punti
        nrighe++;
    }
    Tin_media = Tin_media / (double) nrighe;
    cout << Tin_media << endl;
    ofstream fileout("D:\\EEE\\ROOT lessons\\out.txt");
    fileout << Tin_media ;

    TCanvas *c = new TCanvas(); Dichiarazione finestra grafica
    g->Draw("AP*");
    g->GetXaxis()->SetTimeDisplay(1);
    TDateTime da(2007,01,01,00,00,00);
    gStyle->SetTimeOffset(da.Convert());
    g->GetXaxis()->SetTimeFormat("#splitline{%d\\/%m\\/%y}{%H:%M}");
    g->GetXaxis()->SetNdivisions(10,5,0,kTRUE);
    g->GetXaxis()->SetLabelOffset(0.030);
}

```



```

{
    // inizio della macro

    ifstream filedati("D:\\EEE\\ROOT lessons\\weather_trending.csv");
    if(filedati.fail())
    {
        cout << "Il file non esiste, verifica il percorso!" << endl;
        break;
    }
    string titolo1, titolo2, titolo3, titolo4, titolo5;
    double secondi, Tin, Tout, pressione, run;
    TGraph *g = new TGraph(); Dichiarazione TGraph (vuoto)

    filedati >> titolo1 >> titolo2 >> titolo3 >> titolo4 >> titolo5;

    double Tin_media = 0;
    int nrighe = 0;

    while(filedati >> secondi >> Tin >> Tout >> pressione >> run)
    {
        Tin_media = Tin_media + Tin;
        g->SetPoint(nrighe, secondi, Tout); Assegnazione coordinate dei punti
        nrighe++;
    }
    Tin_media = Tin_media / (double) nrighe;
    cout << Tin_media << endl;
    ofstream fileout("D:\\EEE\\ROOT lessons\\out.txt");
    fileout << Tin_media ;

    TCanvas *c = new TCanvas(); Dichiarazione finestra grafica
    g->Draw("AP*"); Disegna il TGraph
    g->GetXaxis()->SetTimeDisplay(1);
    TDateTime da(2007,01,01,00,00,00);
    gStyle->SetTimeOffset(da.Convert());
    g->GetXaxis()->SetTimeFormat("#splitline{%d\\/%m\\/%y}{%H:%M}");
    g->GetXaxis()->SetNdivisions(10,5,0,kTRUE);
    g->GetXaxis()->SetLabelOffset(0.030);
}

```

```

{
    // inizio della macro

    ifstream filedati("D:\\EEE\\ROOT lessons\\weather_trending.csv");
    if(filedati.fail())
    {
        cout << "Il file non esiste, verifica il percorso!" << endl;
        break;
    }
    string titolo1, titolo2, titolo3, titolo4, titolo5;
    double secondi, Tin, Tout, pressione, run;
    TGraph *g = new TGraph(); Dichiarazione TGraph (vuoto)

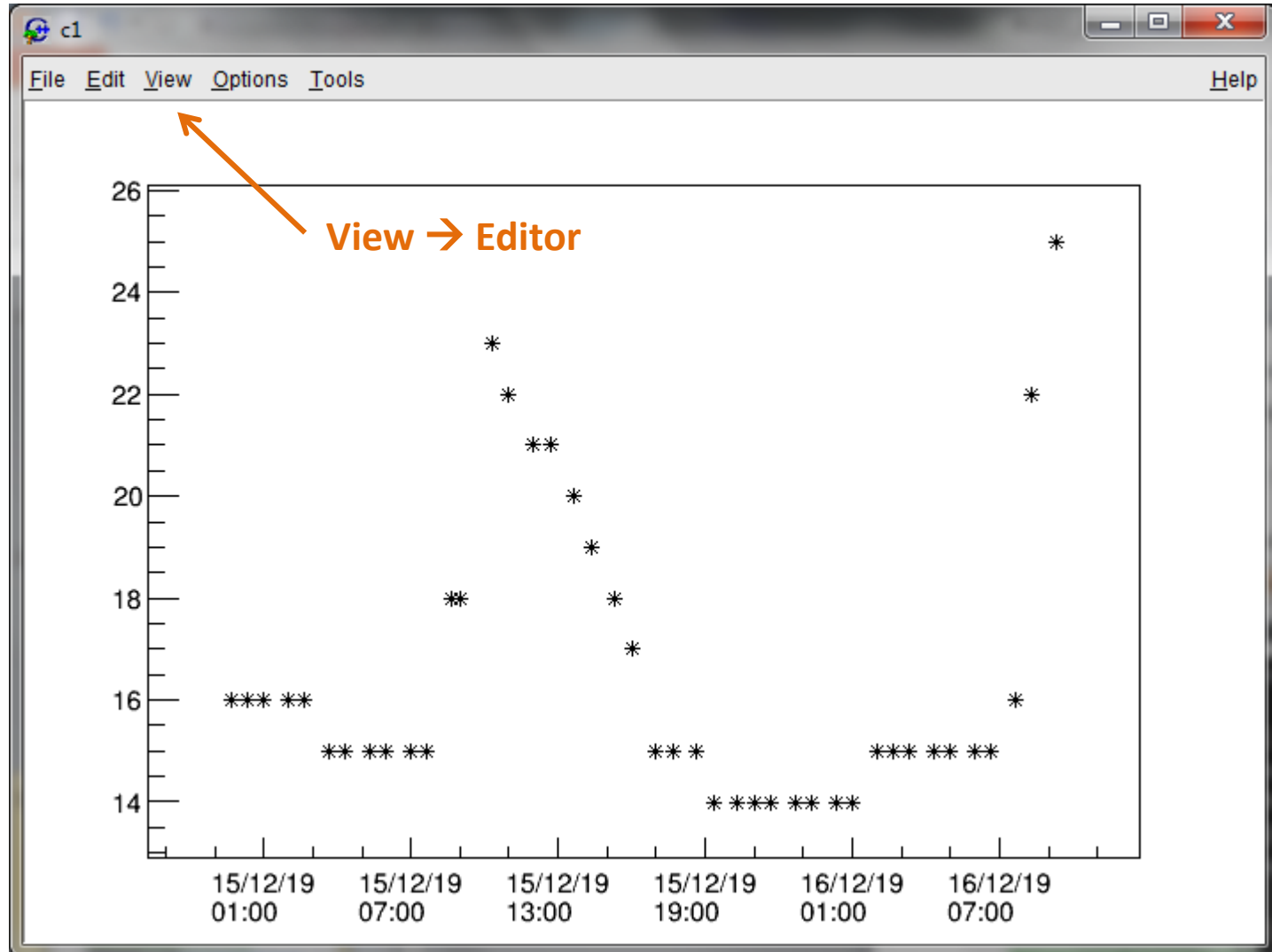
    filedati >> titolo1 >> titolo2 >> titolo3 >> titolo4 >> titolo5;

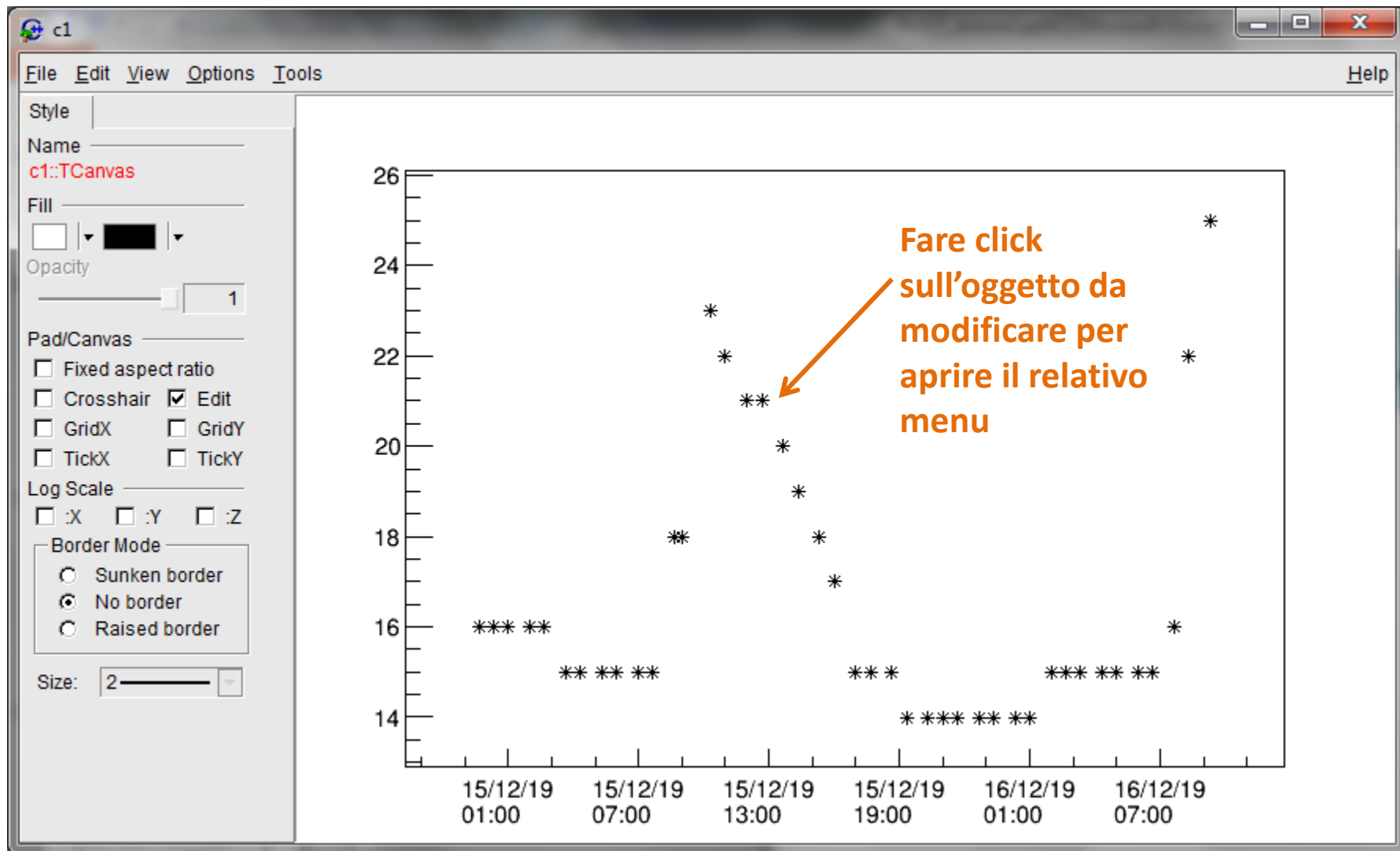
    double Tin_media = 0;
    int nrighe = 0;

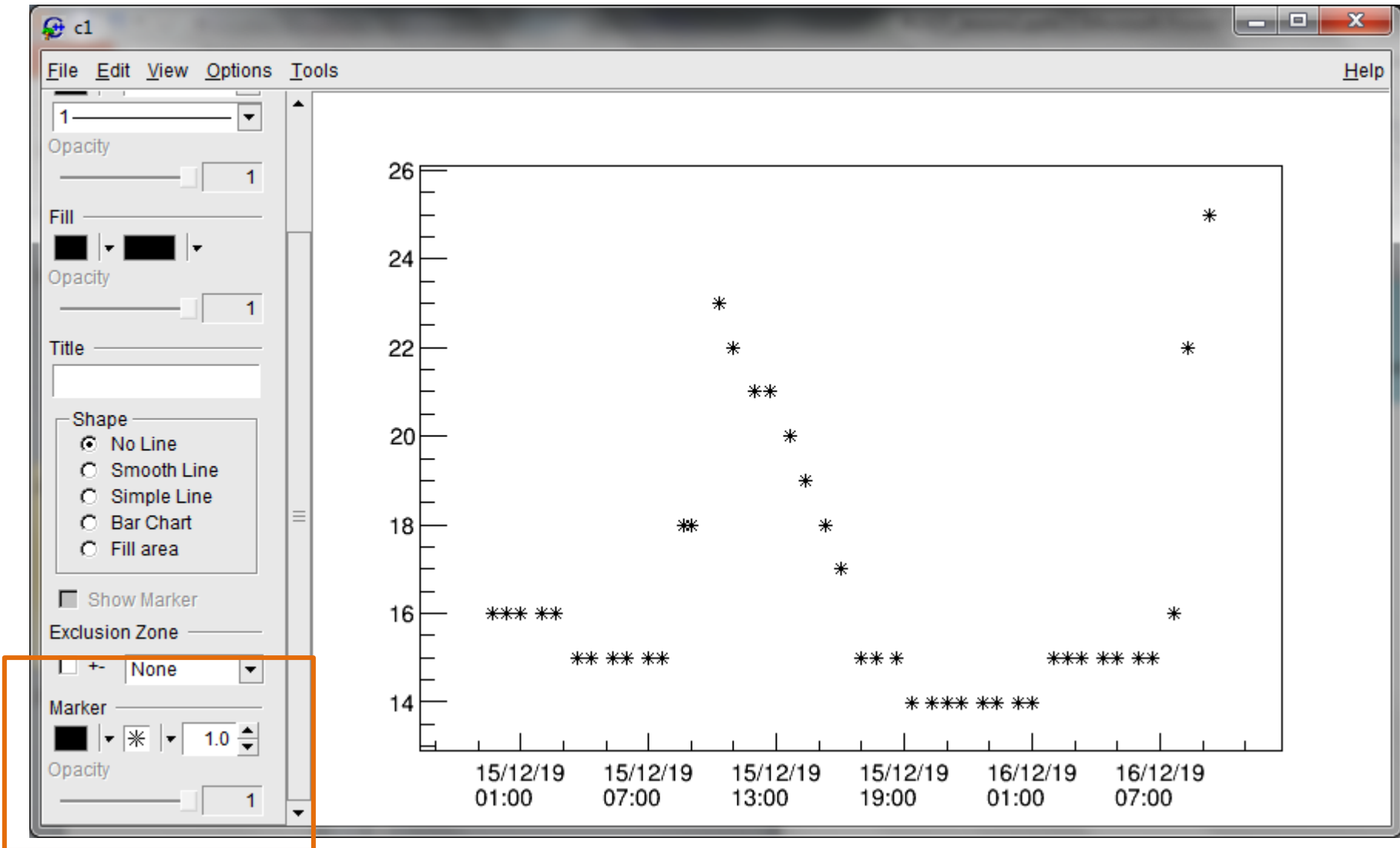
    while(filedati >> secondi >> Tin >> Tout >> pressione >> run)
    {
        Tin_media = Tin_media + Tin;
        g->SetPoint(nrighe, secondi, Tout); Assegnazione coordinate dei punti
        nrighe++;
    }
    Tin_media = Tin_media / (double) nrighe;
    cout << Tin_media << endl;
    ofstream fileout("D:\\EEE\\ROOT lessons\\out.txt");
    fileout << Tin_media ;

    TCanvas *c = new TCanvas(); Dichiarazione finestra grafica
    g->Draw("AP*"); Disegna il TGraph
    g->GetXaxis()->SetTimeDisplay(1); Istruzioni per migliorare la visualizzazione dell'asse
    TDateTime da(2007,01,01,00,00,00); orizzontale (secondi a partire dal 1.1.2007 00:00)
    gStyle->SetTimeOffset(da.Convert());
    g->GetXaxis()->SetTimeFormat("#splitline{%d\\/%m\\/%y}{%H:%M}");
    g->GetXaxis()->SetNdivisions(10,5,0,kTRUE);
    g->GetXaxis()->SetLabelOffset(0.030);
}

```

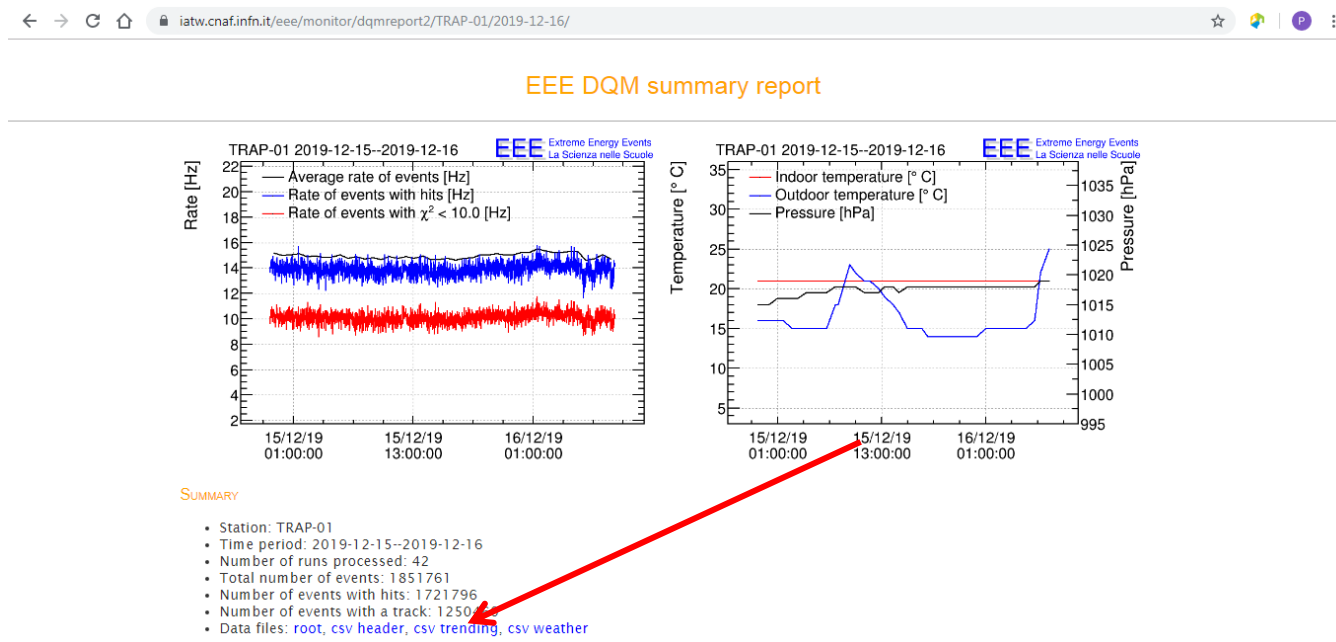






PROPOSTA DI ESERCIZIO/ANALISI

1. Scrivere una macro per leggere i dati presenti nel file csv trending (22 variabili!!!!)



PROPOSTA DI ESERCIZIO/ANALISI

1. Scrivere una macro per leggere i dati presenti nel file csv trending (22 variabili!!!!)
2. Calcolare il valore medio di:
 - RateTrackEvents
 - RateTrackEventsTheta0_10
 - RateTrackEventsTheta10_20
 - ...

PROPOSTA DI ESERCIZIO/ANALISI

1. Scrivere una macro per leggere i dati presenti nel file csv trending (22 variabili!!!!)
2. Calcolare il valore medio di:
 - RateTrackEvents
 - RateTrackEventsTheta0_10
 - RateTrackEventsTheta10_20
 - ...
3. Scrivere le medie su un file di output

PROPOSTA DI ESERCIZIO/ANALISI

1. Scrivere una macro per leggere i dati presenti nel file csv trending (22 variabili!!!!)
2. Calcolare il valore medio di:
 - RateTrackEvents
 - RateTrackEventsTheta0_10
 - RateTrackEventsTheta10_20
 - ...
3. Scrivere le medie su un file di output
4. Realizzare e disegnare un TGraph che contenga i punti (Θ_i , Rate_i)

PROPOSTA DI ESERCIZIO/ANALISI

SUGGERIMENTO:

- Creare un array di 7 variabili (es. $y[7]$) per il calcolo delle medie di:

RateTrackEventsTheta0_10
RateTrackEventsTheta10_20
RateTrackEventsTheta20_30
RateTrackEventsTheta30_40
RateTrackEventsTheta40_50
RateTrackEventsTheta50_60
RateTrackEventsTheta60_90

e un array per i corrispondenti valori di Theta (es. $x[7] = \{5, 15, \dots\}$)

- Infine creare un TGraph con questi dati:

```
TGraph *g = new TGraph (7, x, y);  
g->Draw("AP*");
```

**SALVATE LA MACRO IN UN FILE (all'interno della cartella macros) ED
ESEGUITELA DAL TERMINALE DI ROOT (con il comando .x)**



**PER CHIARIMENTI E DUBBI RIVOLGERSI
AL PROPRIO RESPONSABILE LOCALE**