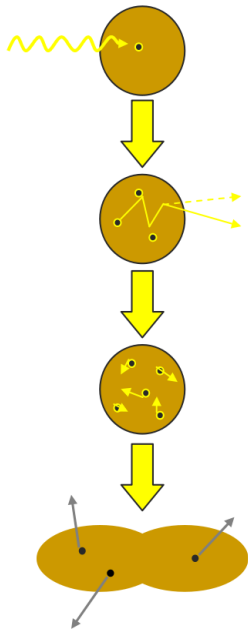


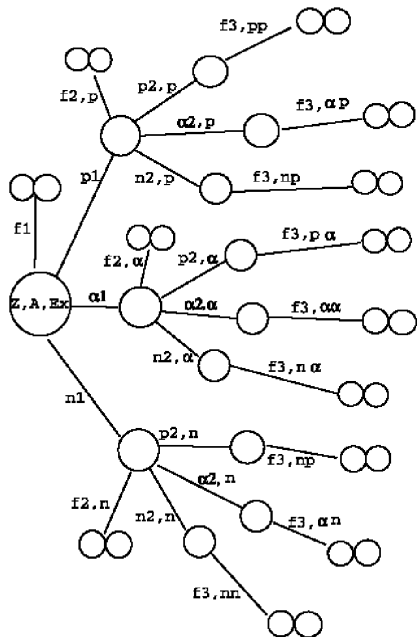
Introdução à Física Nuclear
Trabalho 1 - Fórmula de Massa

22 de agosto de 2012

Código CRISP

- ▶ Simulação de reações nucleares;
 - ▶ Reações entre 5 MeV e 3.5 GeV;
 - ▶ Fissão induzida por prótons e fótons em íons pesados;
 - ▶ Reações de Spallation.
- ▶ Escrito em C++;
- ▶ Usa as bibliotecas da plataforma ROOT (CERN), principalmente para análise de resultados.
- ▶ Simulação em duas etapas:
 - ▶ Cascata intranuclear;
 - ▶ Competição entre evaporação de partículas e fissão.





Plataforma ROOT

- ▶ Análise de dados
 - ▶ Gráficos;
 - ▶ Histogramas;
 - ▶ Funções matemáticas (funções especiais, densidade de probabilidade, distribuições cumulativas);
 - ▶ Álgebra linear;
 - ▶ Geometria;
 - ▶ Minuit.
- ▶ Escrito em C++ (modo interativo e compilado);
- ▶ <http://root.cern.ch/root/html/ClassIndex.html>

Exemplo: Gráfico com barras de erro

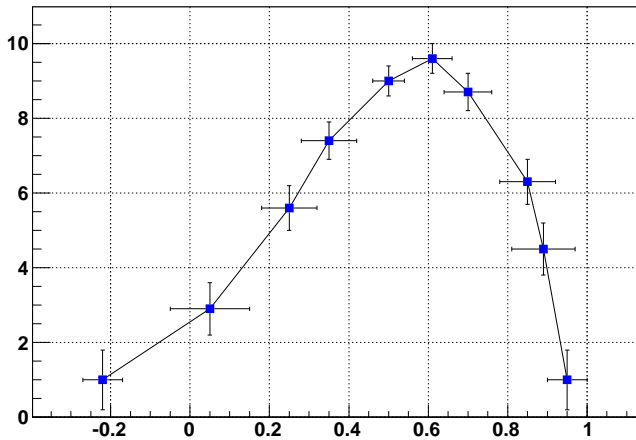
```
1 void gerrors() {
2   //Draw a graph with error bars
3   // To see the output of this macro, click begin_html <a href="gif/gerrors.gif">her
4   //Author: Rene Brun
5
6   TCanvas *c1 = new TCanvas("c1","A Simple Graph with error bars",200,10,700,500);
7
8   c1->SetFillColor(10);
9   c1->SetGrid();
10  c1->GetFrame()->SetFillColor(10);
11  c1->GetFrame()->SetBorderSize(0);
12
13  const Int_t n = 10;
14  Float_t x[n] = {-0.22, 0.05, 0.25, 0.35, 0.5, 0.61,0.7,0.85,0.89,0.95};
15  Float_t y[n] = {1,2.9,5.6,7.4,9,9.6,8.7,6.3,4.5,1};
16  Float_t ex[n] = {.05,.1,.07,.07,.04,.05,.06,.07,.08,.05};
17  Float_t ey[n] = {.8,.7,.6,.5,.4,.4,.5,.6,.7,.8};
18  TGraphErrors *gr = new TGraphErrors(n,x,y,ex,ey);
19  gr->SetTitle("TGraphErrors Example");
20  gr->SetMarkerColor(4);
21  gr->SetMarkerStyle(21);
22  gr->Draw("ALP");
23
24  c1->Update();|
25 }
```

```
evandro@thavma: ~
Arquivo  Editar  Ver  Terminal  Ajuda
evandro@thavma:~$ root
*****
*
*           W E L C O M E  to  R O O T           *
*
*   Version  5.28/00d           7 May 2011      *
*
*   You are welcome to visit our Web site      *
*           http://root.cern.ch                *
*
*****

ROOT 5.28/00d (tags/v5-28-00d@39141, May 19 2011, 12:47:00 on linuxx8664gcc)

CINT/ROOT C/C++ Interpreter version 5.18.00, July 2, 2010
Type ? for help. Commands must be C++ statements.
Enclose multiple statements between { }.
root [0] .x gerrors.C
```

TGraphErrors Example



Minuit

```
15 #include <iostream.h>
16
17 #include <TR00T.h>
18 #include <TMinuit.h>
19
20 const int iNum = 5;
21 Float_t z[5],x[5],y[5],errorz[5];
22
23 // this is the function used for the fit
24 // par: vector with the fit parameters
25 Double_t fit_function(float x,float y,Double_t *par)
26 {
27     double value=( (par[0]*par[0])/(x*x)-1)/ ( par[1]+par[2]*y-par[3]*y*y);
28     return value;
29 }
30
31 void calc_chi_square(Int_t &npar, Double_t *gin, Double_t &f, Double_t *par, Int_t iflag)
32 {
33     //calculate chisquare
34     double chisq = 0;
35     for (int i=0;i<iNum; i++) {
36         // chi square is the quadratic sum of the distance from the point to the function
37         // weighted by its error
38         double delta = (z[i]-fit_function(x[i],y[i],par))/errorz[i];
39         chisq += delta*delta;
40     }
41     f = chisq;
42 }
```

```
43
44 int Minuit_examp()
45 {
46     // The z values
47     z[0]=1;
48     z[1]=0.96;
49     z[2]=0.89;
50     z[3]=0.85;
51     z[4]=0.78;
52     // The errors on z values
53     Float_t error = 0.01;
54     errorz[0]=error;
55     errorz[1]=error;
56     errorz[2]=error;
57     errorz[3]=error;
58     errorz[4]=error;
59     // the x values
60     x[0]=1.5751;
61     x[1]=1.5825;
62     x[2]=1.6069;
63     x[3]=1.6339;
64     x[4]=1.6706;
65     // the y values
66     y[0]=1.0642;
67     y[1]=0.97685;
68     y[2]=1.13168;
69     y[3]=1.128654;
70     y[4]=1.44016;
71
```

```
72 //initialize TMinuit with a maximum of 5 params
73 TMinuit *ptMinuit = new TMinuit(5);
74 //
75 // select verbose level:
76 // default : (58 lines in this test)
77 // -1 : minimum ( 4 lines in this test)
78 // 0 : low (31 lines)
79 // 1 : medium (61 lines)
80 // 2 : high (89 lines)
81 // 3 : maximum (199 lines in this test)
82 //
83 ptMinuit->SetPrintLevel();
84 // set the user function that calculates chi_square
85 //(the value to minimize)
86 ptMinuit->SetFCN(calc_chi_square);
87
88 Double_t arglist[10];
89 Int_t ierflg = 0;
90
91 arglist[0] = 1;
92 ptMinuit->mnexcm("SET ERR", arglist ,1,ierflg);
93
94 // Set starting values and step sizes for parameters
95 static Double_t vstart[4] = {3, 1, 0.1, 0.01};
96 static Double_t step[4] = {0.1, 0.1, 0.01, 0.001};
97 ptMinuit->mnparm(0, "a1", vstart[0], step[0], 0,0,ierflg);
98 ptMinuit->mnparm(1, "a2", vstart[1], step[1], 0,0,ierflg);
99 ptMinuit->mnparm(2, "a3", vstart[2], step[2], 0,0,ierflg);
100 ptMinuit->mnparm(3, "a4", vstart[3], step[3], 0,0,ierflg);
```

```

102 // Now ready for minimization step
103 arglist[0] = 500;
104 arglist[1] = 1.;
105 ptMinuit->mnexcm("MIGRAD", arglist ,2,ierflg);
106
107 // Print results
108 Double_t amin,edm,errdef;
109 Int_t nvar,npvar,icstat;
110 ptMinuit->mnstat(amin,edm,errdef,nvar,npvar,icstat);
111 //void mnstat(Double_t &fmin, Double_t &fedm, Double_t &errdef, Int_t &npari,
112 //            [Int_t &nparx, Int_t &istat)
113 //**.*-.*-.*Returns concerning the current status of the minimization*-*.*-.*
114 //**.*
115 //**.*      User-called
116 //**.*      Namely, it returns:
117 //**.*      FMIN: the best function value found so far
118 //**.*      FEDM: the estimated vertical distance remaining to minimum
119 //**.*      ERRDEF: the value of UP defining parameter uncertainties
120 //**.*      NPARI: the number of currently variable parameters
121 //**.*      NPARX: the highest (external) parameter number defined by user
122 //**.*      ISTAT: a status integer indicating how good is the covariance
123 //**.*      matrix: 0= not calculated at all
124 //**.*      1= approximation only, not accurate
125 //**.*      2= full matrix, but forced positive-definite
126 //**.*      3= full accurate covariance matrix
127 //**.*-.*-.*-.*-.*-.*-.*-.*-.*-.*-.*-.*-.*-.*-.*-.*-.*-.*-.*-.*-.*-.*-.*
128 return 0;
129 }

```

Lista de comandos do Minuit:

<http://wwwasdoc.web.cern.ch/wwwasdoc/minuit/node18.html>