PGPLOT

PHYS2G03

McMaster University

Plotting Data

- Researchers employ two main ways of plotting data graphically:
- 1) Generate a file and use a separate package on the file e.g. IDL, gnuplot, TOPCAT, spreadsheet software, python
- 2) Call graphics routines directly using a library of routines

PGPLOT

- PGPLOT is a library of routines callable from C/C++ and FORTRAN (77 or 90)

- It produces very nice graphs, contour plots, surface plots and other graphics

PGPLOT Library

- PGPLOT has to be downloaded and installed from the website, it is freeware
- It is installed on phys-ugrad and should just work
- On some systems PGPLOT needs environment variables. The current phys-ugrad does not.

- e.g.

  setenv PGPLOT_DIR /usr/share/pgplot

  setenv LD_LIBRARY_PATH /opt/intel_fc_80/lib:/usr/local/lib:/2/local/lib:/usr/X11R6/lib:.

PGPLOT Devices

When PGPLOT runs it offers a list of devices: e.g. /XWINDOW, /GIF, /CPS

Just enter /xw

(shorthand for /XWINDOW)

PGPLOT

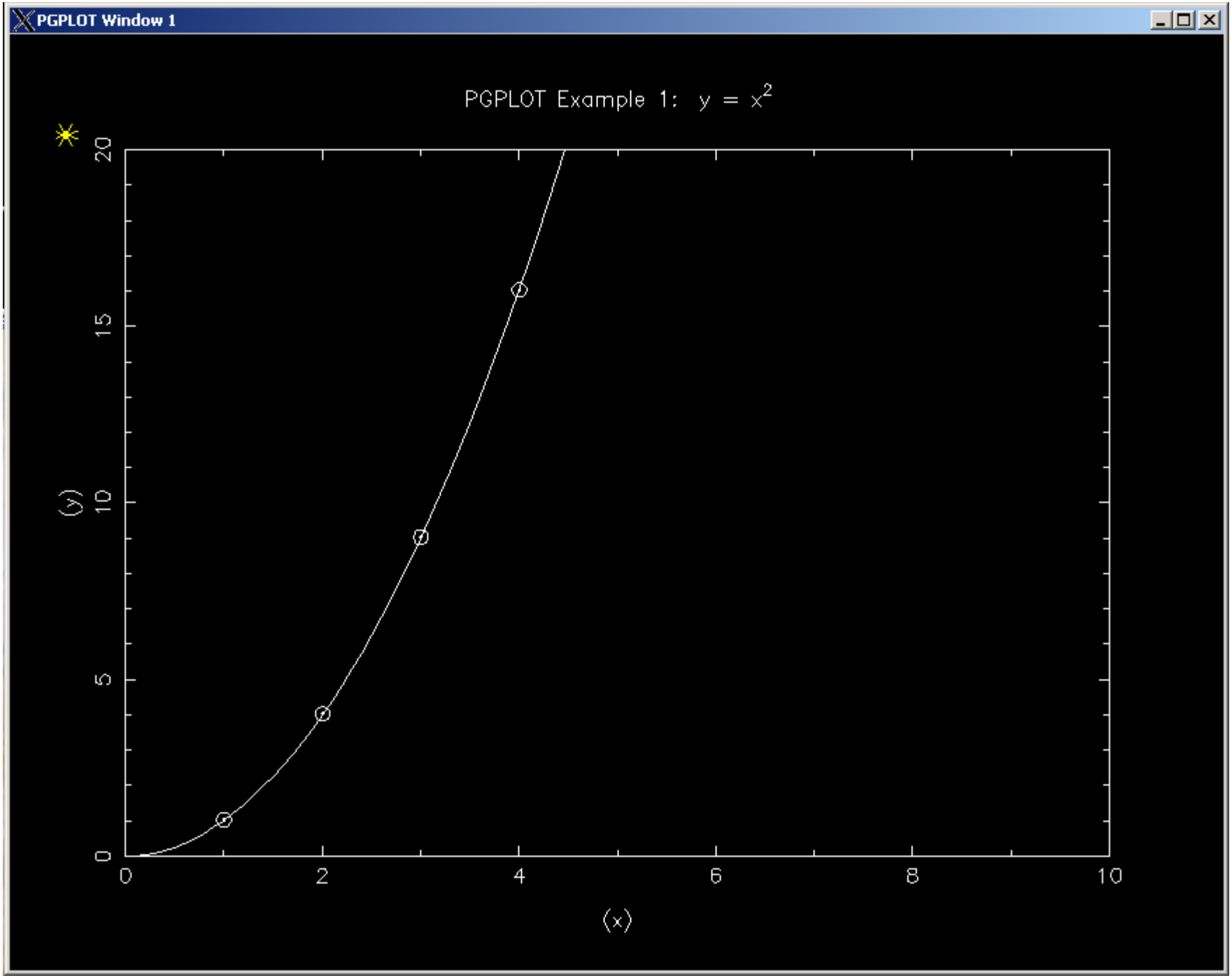
/home/2G03/pgplot/pgdemo1

Just hit return for the next plot

There are 9 demo programs

pgdemo1 … pgdemo9

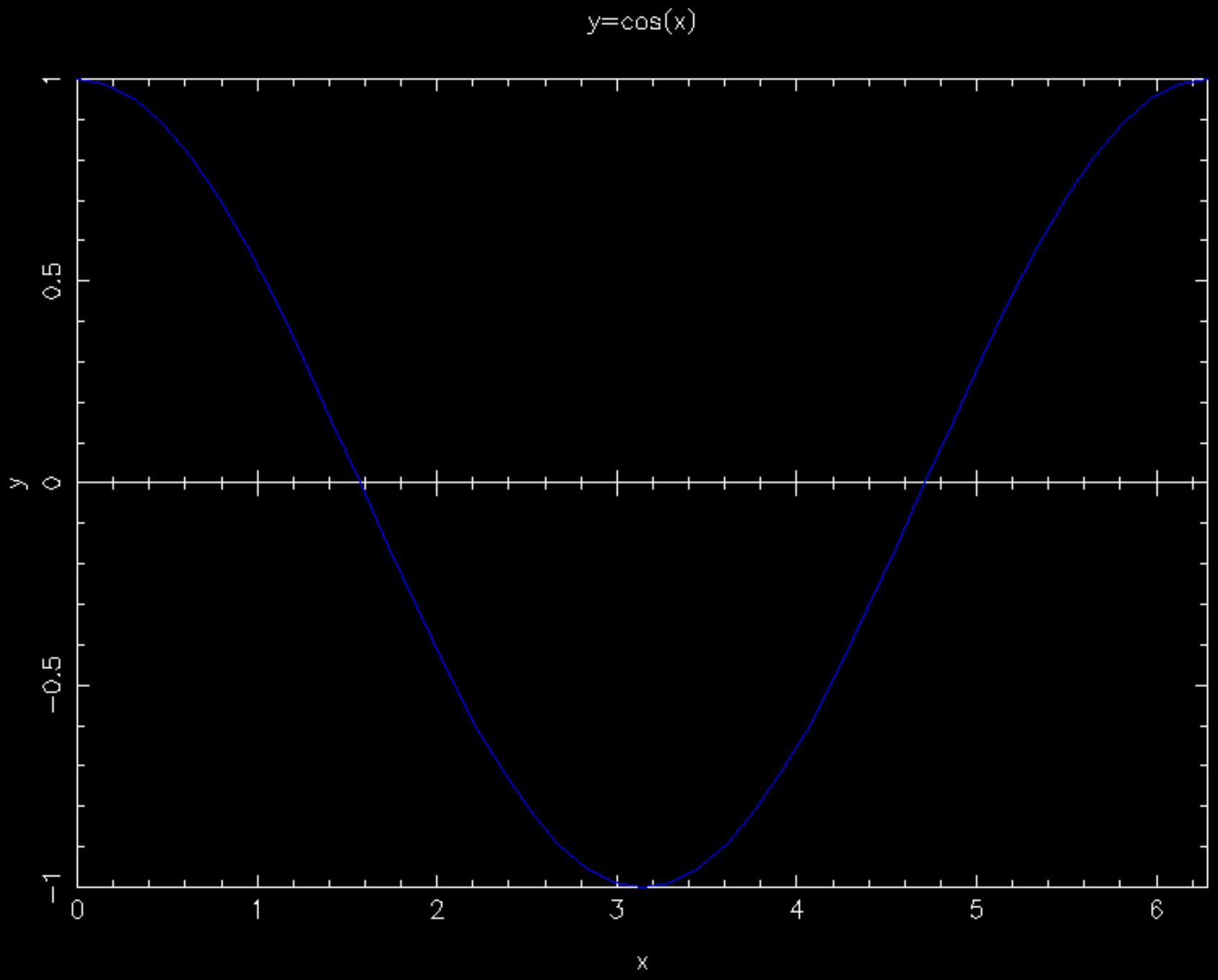
If you want to know if pgplot can do something it is probably in one of the demo plots to find out


PGPLOT Window 1
PGPLOT Example 1: y = x2
(y)
0
5
10
15
20
0
2
4
6
8
10
(x)

C/C++ code

Uses PGPLOT to plot y=cos(x)

```
cp –r /home/2G03/ploty ~/
cd ~/ploty
make
ploty
```


y=cos(x)
1
0.5
0
−0.5
−1
y
0
1
2
3
4
5
6
x

```
#include <cmath>
#include "cpgplot.h"

int main()
{
  //-----------------------------------------------------
  // This program uses pgplot to plot y = cos(x)
  //-----------------------------------------------------
  const int nintervals = 40;
  float x[nintervals+1], y[nintervals+1];
  float pi, dx;
  int i;

  // Open a plot window
  if (!cpgopen("/XWINDOW")) return 1;

  // Set-up plot axes
 // M_PI is defined in cmath
  cpgenv(0.,2*M_PI,-1.,1.,0,1);
  // Label axes
  cpglab("x", "y", "y=cos(x)");
```

ploty.cpp

```
 dx = 2*M_PI/nintervals;

  // Change plot colour to colour 2 (red)
  cpgsci(4);

  // Compute the function at the points
  for (i=0;i<=nintervals;i++) {
    x[i] = dx*i;
    y[i] = cos(x[i]);
  }

  // Plot the curve
  cpgline(nintervals+1,x,y);

  // Pause and then close plot window
  cpgclos();

}
```

Compiling with PGPLOT

- PGPLOT is a non-standard library so the linker has to be told where to find the routines

PGPLOT library

PGPLOT C/C++ library

```
ploty: ploty.o
	c++ -o ploty ploty.o -ltrapfpe -lpgplot -lcpgplot -lX11

ploty.o: ploty.cpp
	c++ -c ploty.cpp
```

Xwindows library

Key PGPLOT routines

Add prototypes for pgplot functions

```
#include "cpgplot.h”
```

Open a plot window (Xwindow)

```
 cpgopen("/XWINDOW”)
```

Set-up plot axes

```
  cpgenv(0.,2*M_PI,-1.,1.,0,1);
```

Label axes

```
  cpglab("x", "y", "y=cos(x)");
```

Change plot colour to colour 4 (BLUE)

```
  cpgsci(4);
```

Plot two arrays x and y as a curve

```
  cpgline(nintervals+1,x,y);
```

Pause and then close plot window

```
  cpgclos();
```

Key PGPLOT routines

Set-up plot axes

```
cpgenv(0.,2*M_PI,-1.,1.,0,1);
```

1st, 2nd arguments x-axis min and max

3rd, 4th arguments y-axis min and max

5th argument: use same scaling on axes 0=no 1=yes

6th argument: axis style: box plus axes drawn

Label axes

```
cpglab("x", "y", "y=cos(x)");
```

1st, 2nd arguments, x and y axis labels

3rd argument over plot title

Key PGPLOT routines

Change plot colour (0=white, 1 black,…

cpgsci(col);

Plot two arrays x and y as a curve

cpgline(n,x,y);

1st argument: number of points

2nd , 3rd arguments, x and y arrays

MUST be float NOT double

Plot two arrays x and y as a points

cpgpt(n,x,y,sym);

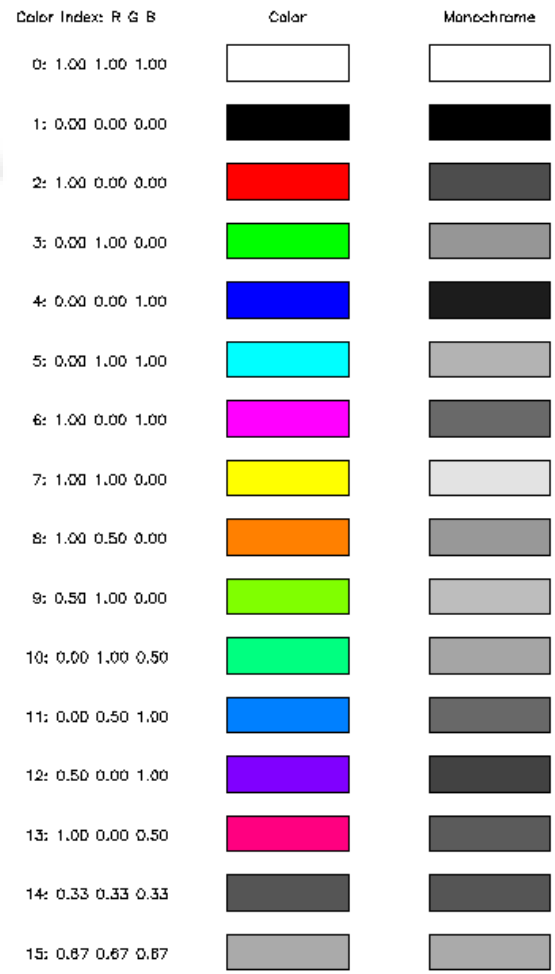
1st argument: number of points

2nd , 3rd arguments, x and y arrays

MUST be float NOT double

4th argument: symbol to use 0=box, 1=dot, 2=+, 3=*, 4=circle …

must be int



Documentation

PGPLOT full documentation/download:

- http://www.astro.caltech.edu/~tjp/pgplot/

  Have a look at this website now – it lists the function for Fortran first

C/C++ is described here.
http://www.astro.caltech.edu/~tjp/pgplot/cbinding.html

Most functions are the same, but lower case with a “c” in front for C/C++

e.g. pgline(n,x,y) → cpgline(n,x,y); // C++ version

www.astro.caltech.edu/~tjp/pgplot/subroutines.html#PGENV

Homepage - PHYSICS 2003Sci | PGPLOT Subroutine Descriptions

Not secure | star-www.dur.ac.uk/~ns/pgplot.html#PGENV

PGENV -- set window and viewport and draw labeled frame

```
      SUBROUTINE PGENV (XMIN, XMAX, YMIN, YMAX, JUST, AXIS)
      REAL XMIN, XMAX, YMIN, YMAX
      INTEGER JUST, AXIS
```

Set PGPLOT "Plotter Environment". PGENV establishes the scaling for subsequent calls to PGPT, PGLINE, etc. The plotter is advanced to a new page or panel, clearing the screen if necessary. If the "prompt state" is ON (see PGASK), confirmation is requested from the user before clearing the screen. If requested, a box, axes, labels, etc. are drawn according to the setting of argument AXIS.

```
Arguments:
 XMIN   (input)  : the world x-coordinate at the bottom left corner
                   of the viewport.
 XMAX   (input)  : the world x-coordinate at the top right corner
                   of the viewport (note XMAX may be less than XMIN).
 YMIN   (input)  : the world y-coordinate at the bottom left corner
                   of the viewport.
 YMAX   (input)  : the world y-coordinate at the top right corner
                   of the viewport (note YMAX may be less than YMIN).
 JUST   (input)  : if JUST=1, the scales of the x and y axes (in
                   world coordinates per inch) will be equal,
                   otherwise they will be scaled independently.
 AXIS   (input)  : controls the plotting of axes, tick marks, etc:
     AXIS = -2 : draw no box, axes or labels;
     AXIS = -1 : draw box only;
     AXIS =  0 : draw box and label it with coordinates;
     AXIS =  1 : same as AXIS=0, but also draw the
                 coordinate axes (X=0, Y=0);
     AXIS =  2 : same as AXIS=1, but also draw grid lines
                 at major increments of the coordinates;
     AXIS = 10 : draw box and label X-axis logarithmically;
     AXIS = 20 : draw box and label Y-axis logarithmically;
     AXIS = 30 : draw box and label both axes logarithmically.
```

For other axis options, use routine PGBOX. PGENV can be persuaded to call PGBOX with additional axis options by defining an environment parameter PGPLOT_ENVOPT containing the required option codes.

```
Examples:
  PGPLOT_ENVOPT=P       | draw Projecting tick marks
  PGPLOT_ENVOPT=I       | Invert the tick marks
  PGPLOT_ENVOPT=IV      | Invert tick marks and label y Vertically
```

PGERAS -- erase all graphics from current page

```
      SUBROUTINE PGERAS
```

Erase all graphics from the current page or panel.

```
Arguments: none
```

Differences in C/C++ pgplot

C/C++ is described here

http://www.astro.caltech.edu/~tjp/pgplot/cbinding.html

Some function arguments are a little different because Fortran passes by reference but C/C++ does not unless you use a pointer explicitly

This only applies to functions that return data such as the cursor locating function, PGCURS:

e.g. FORTRAN: pgcurs(x,y,ch)

C/C++: cpgcurs(&x,&y,&ch);

Examples on phys-ugrad

- phys-ugrad:

  /home/2G03/pgplot Demos (Fortran)

  /home/2G03/ploty C/C++ code demo

see also mandel code for Mandel or lorenz

  /home/2G03/mandel/MandelMain.cpp

  /home/2G03/lorenz/lorenz.cpp