授業資料A4用紙2枚（4面7頁）
スクリーン前椅子＆以下のURLで

http://comp.cs.ehime-u.ac.jp/~okano/computation/

計算科学特論

差分法演習

linux環境 or Windows環境にログインして
授業情報ウェブサイトにアクセスしてください

URL: http://comp.cs.ehime-u.ac.jp/
~okano/computation/

今日の予定

今日のテーマ:

Laplace方程式の境界値問題に差分法を適用し、授業で説明した解法を試してみます。

具体的には、Scilab (サイラボ)のプログラムで差分法のサンプルプログラムを動作させて、その結果を確認します。

今日の予定

Scilab練習

サンプルプログラム実行と解析のヒント

Scilab練習

Scilab,　A Free Scientific Software Package

Scilab は工学・理学での応用に役立つ強力でオープンな計算環境を提供する数値計算のためのソフトウェアパッケージです。Scilabには何百もの数学関数が含まれ、加えて、様々なプログラミング言語(C, C++, Fortran・・・)で作成されたプログラムを必要に応じて取り込む機能を備えています。洗練されたデータ構造(リスト、多項式、有理関数、線形システム・・・)を扱うことができる仕組みと、インタプリタ、高度なプログラミング言語を備えています。

Scilabホームページ (http://www.scilab.org)から抜粋・直訳

よく似たソフトウェアに MATLAB や octave などがあります。scilab や octaveには、商用で非常に広く使われている MATLAB に対するフリー・オープンな代替品としての側面があります。ただし、スクリプト言語の仕様は完全に互換ではありません。また、scilabは積極的に独自の機能・仕様を採用し、MATLABからは徐々に離れつつあると言われています。その一方で octave はユーザインタフェースを含めてそっくりになるモードを備える等、MATLAB 互換路線を継続するようです。

※ 情報工学科計算機室ではMATLAB/octaveも利用できます。興味があれば試してみてください。

多くの方にとって初出の環境だと思いますので、はじめにScilab自身についての演習をします。

配布資料（印刷物）に沿って説明しますので、手元のPCで試しながら説明を聞いてください。

サンプルスクリプト解説

サンプルスクリプト

授業情報ウェブサイトからダウンロードしてください。

http://comp.cs.ehime-u.ac.jp/~okano/computation/

サンプルスクリプトのzipファイルには以下の5つのスクリプトが含まれています。

script.sce　　scilabスクリプトサンプル

laplace1.sce
laplace2.sce　} Laplace方程式の境界値問題の解法

heateq1.sce
heateq2.sce　} 熱伝導方程式の初期値問題の解法

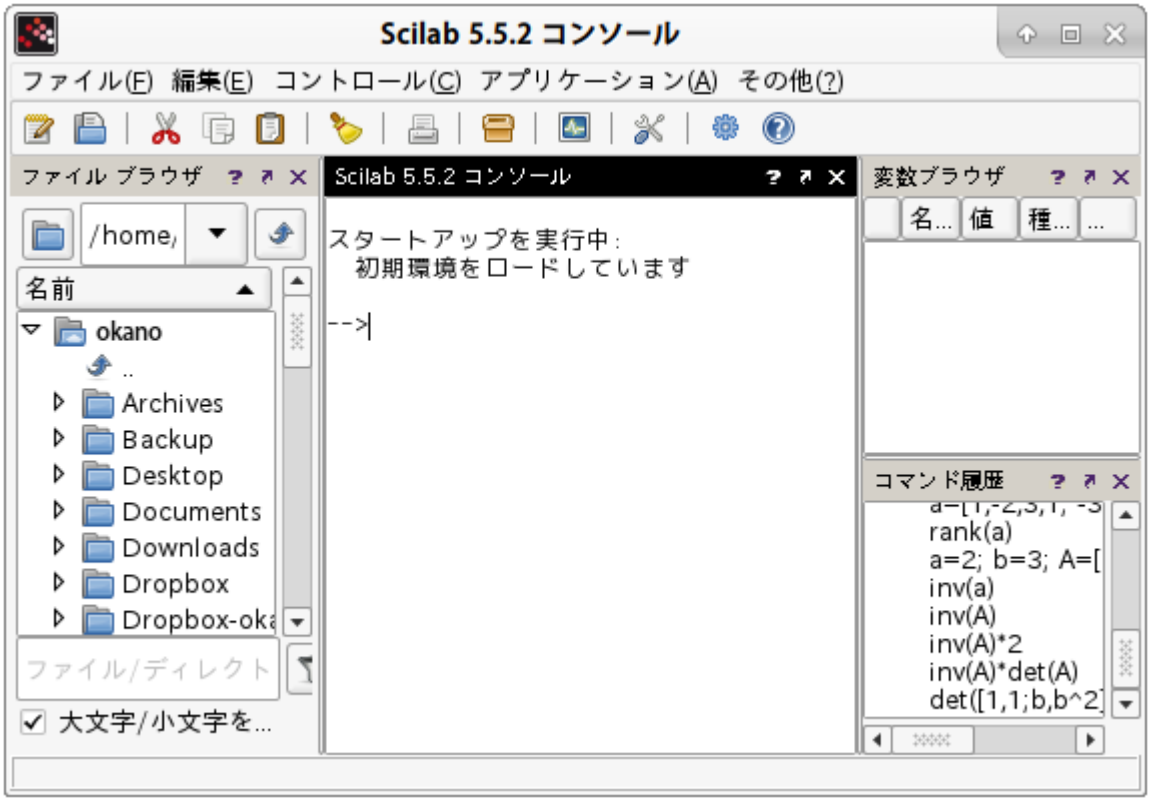
サンプルスクリプト解説

scilabスクリプトサンプル（script.sce）を読み込んで実行してみましょう。

scilab起動直後のウィンドウ（右図）から説明します。

※ 右図はコンソールパネルにfocusがある状態です。メニューやアイコンボタンの様子が異なる場合は、パネル上端の「Scilab～コンソール」をクリックしてください。

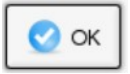
※ Scilabの版/OSによって若干表示が異なります。



サンプルスクリプト解説

scilabスクリプトサンプル（script.sce）を読み込んで実行してみましょう。

「ファイルを開く」ボタン  を押すと「SciNotesで開くファイルを選択」ウィンドウが開きます。

ファイル一覧から目的のスクリプトを選び「OK」ボタン 

を押します。

サンプルスクリプト解説

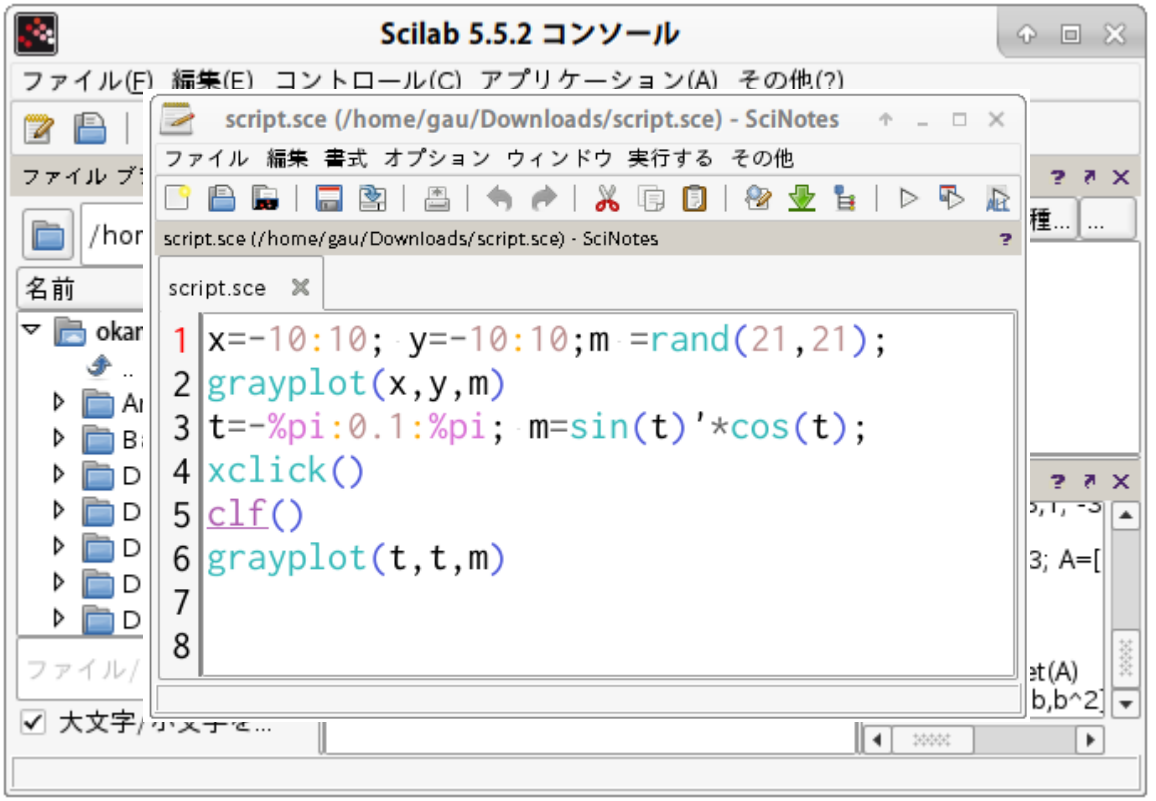
scilabスクリプトサンプル（script.sce）を読み込んで実行してみましょう。

「ファイルを開く」ボタン を押すと「SciNotesで開くファイルを選択」ウィンドウが開きます。

ファイル一覧から目的のスクリプトを選び「OK」ボタン を押します。

「SciNotesで～」とあった通り、SciNotesウィンドウが開きます。

※ 右図はscript.sceを開いた場合です。



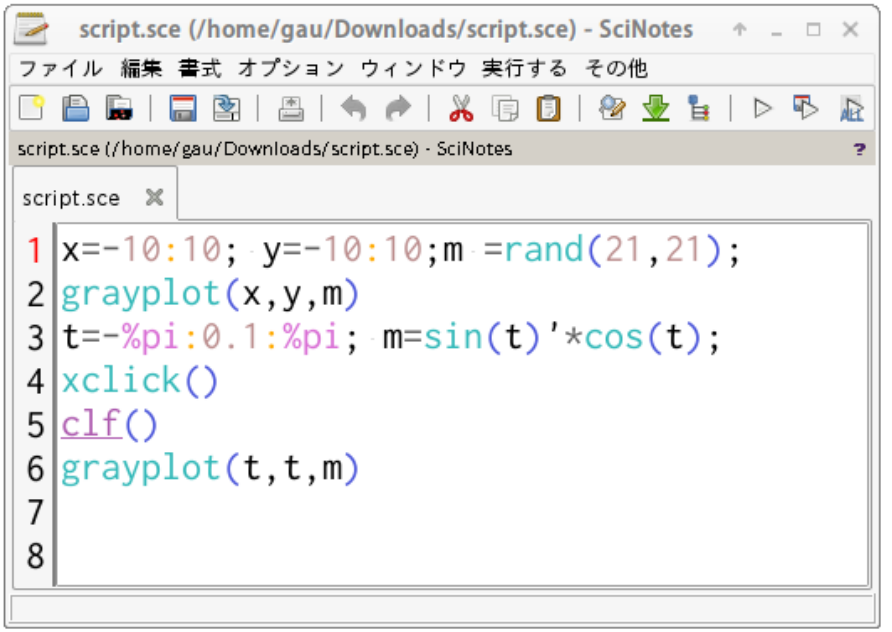
サンプルスクリプト解説

scilabスクリプトサンプル（script.sce）を読み込んで実行してみましょう。

SciNotesはScilab標準のスクリプト開発エディタです。

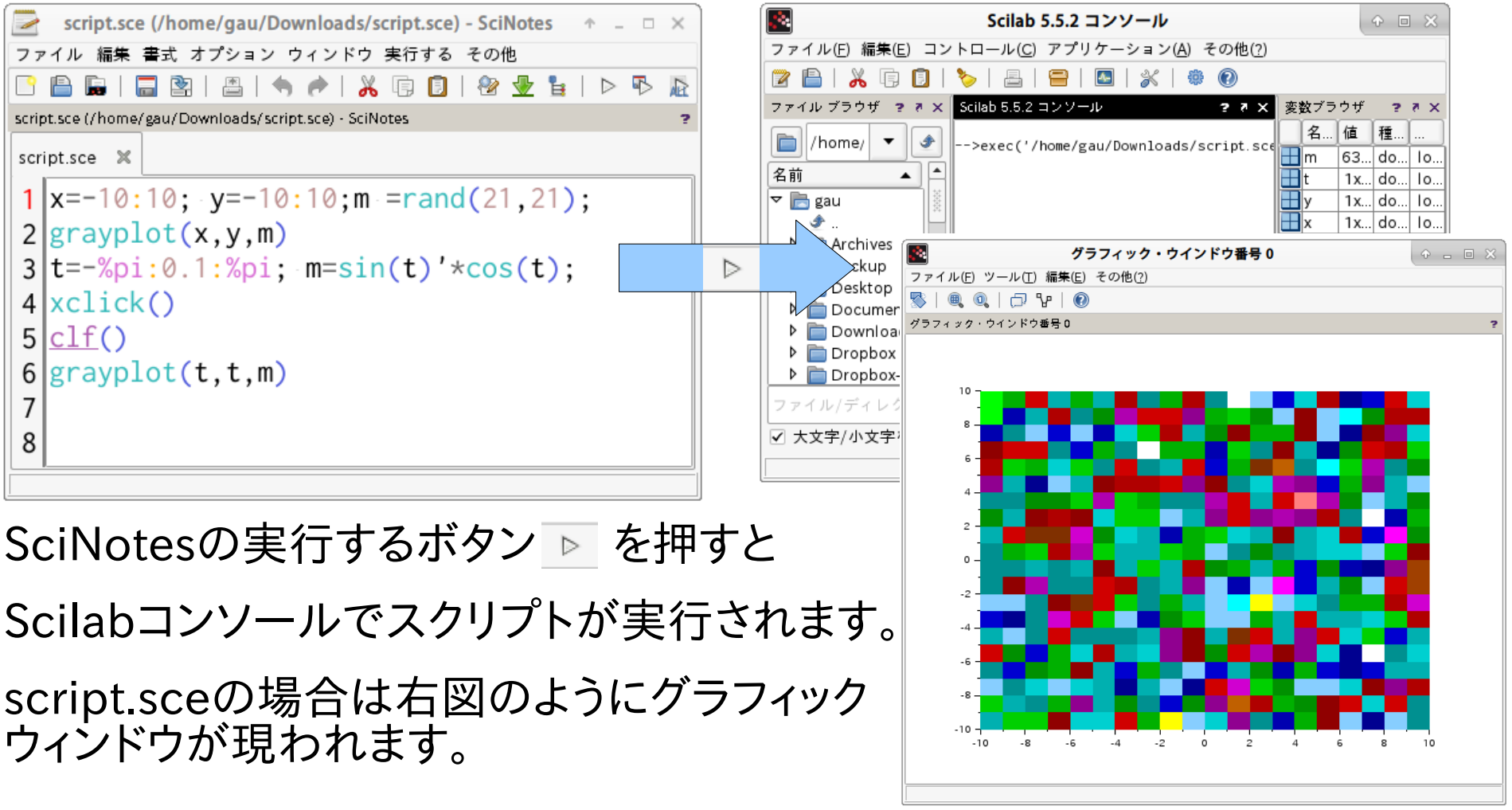
エディタ機能を利用してスクリプトファイルを作成し、Scilabコンソールで実行させることができます。

スクリプト（sript.sce）を右図のように読み込んだ状態で実行するためには、
　　実行するボタン ▷
を押すと、開いているスクリプトがコンソールで実行されます。



サンプルスクリプト解説

scilabスクリプトサンプル（script.sce）を読み込んで実行してみましょう。



SciNotesの実行するボタン ▷ を押すと

Scilabコンソールでスクリプトが実行されます。

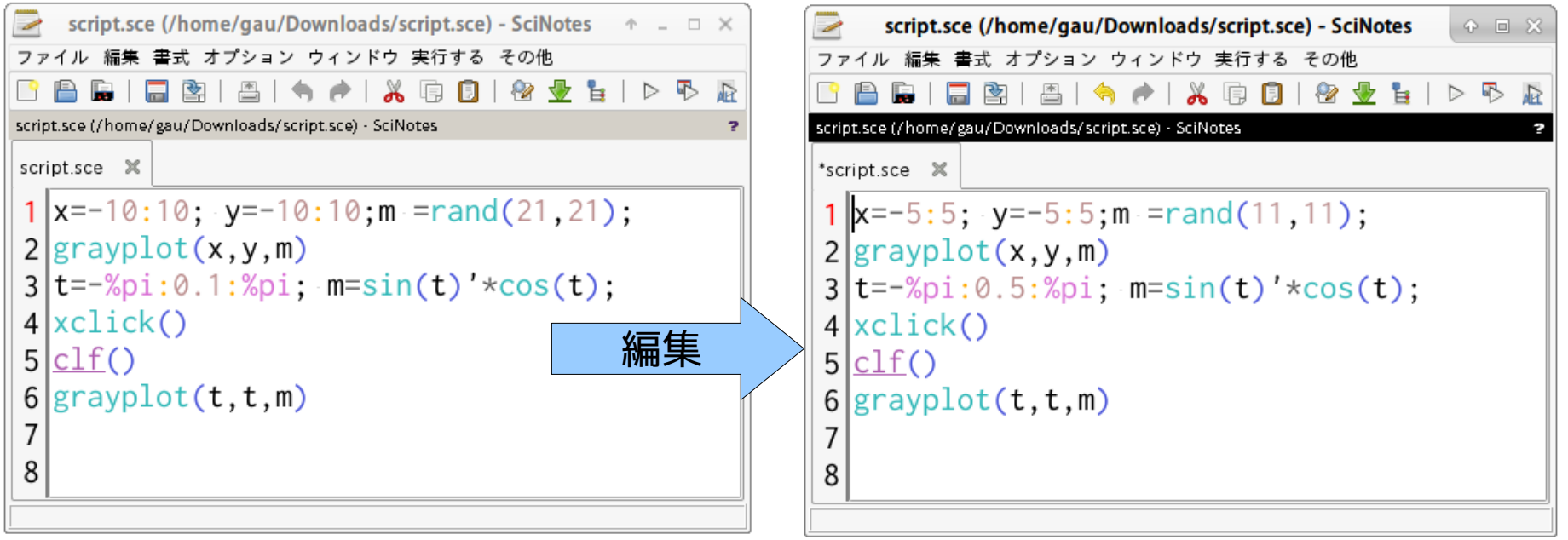
script.sceの場合は右図のようにグラフィックウィンドウが現われます。

サンプルスクリプト解説

scilabスクリプトサンプル（script.sce）を書き換えて実行してみましょう。

script.sce中の数字を下図のように変更して実行してみましょう。
（1行目と3行目の数字に変更があります。）

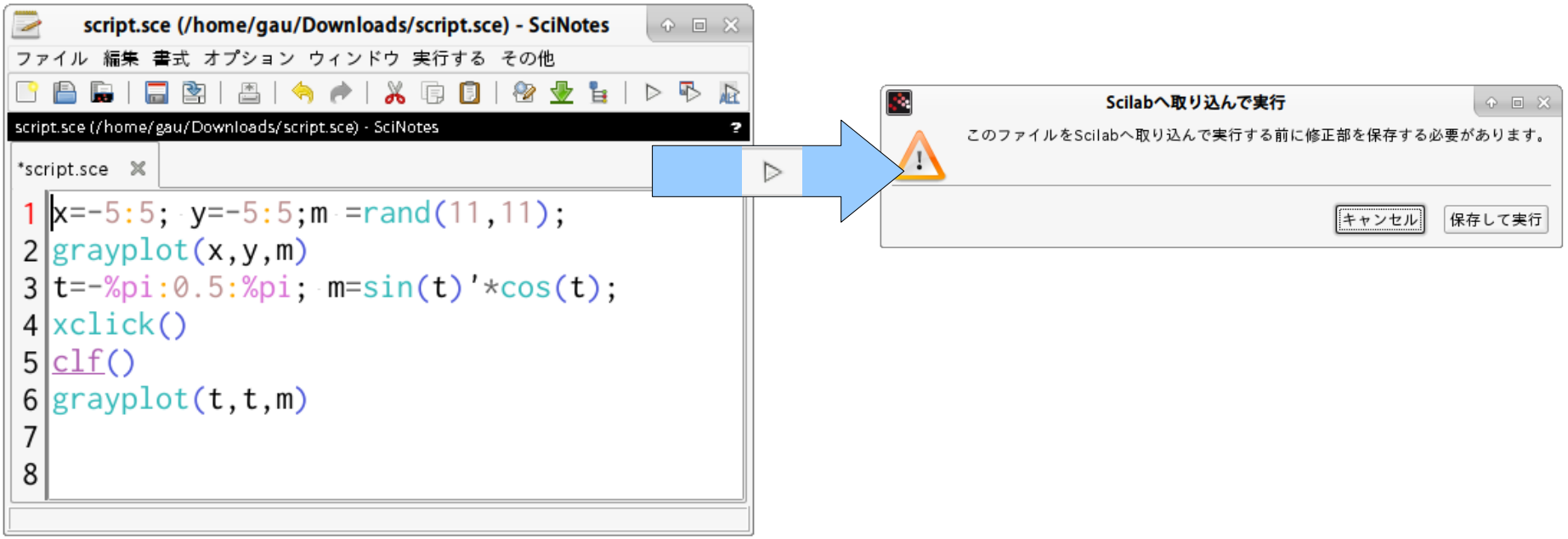
実行結果は期待通りのものでしたか？



サンプルスクリプト解説

scilabスクリプトサンプル（script.sce）を書き換えて実行してみましょう。

script.sceを変更してから実行するボタン ▷ を押すと警告が表示されます。



実行するボタン ▷ の代わりに保存して実行ボタン を使うことで警告を出さずに、変更したスクリプトを実行することができます。

SciNotesはScilabコンソールに対してスクリプトファイルを読み込みを指示することで処理を実現しているため、このような仕組みになっています。

サンプルスクリプト解説

SciNotesを使って他のスクリプトを実行しましょう。

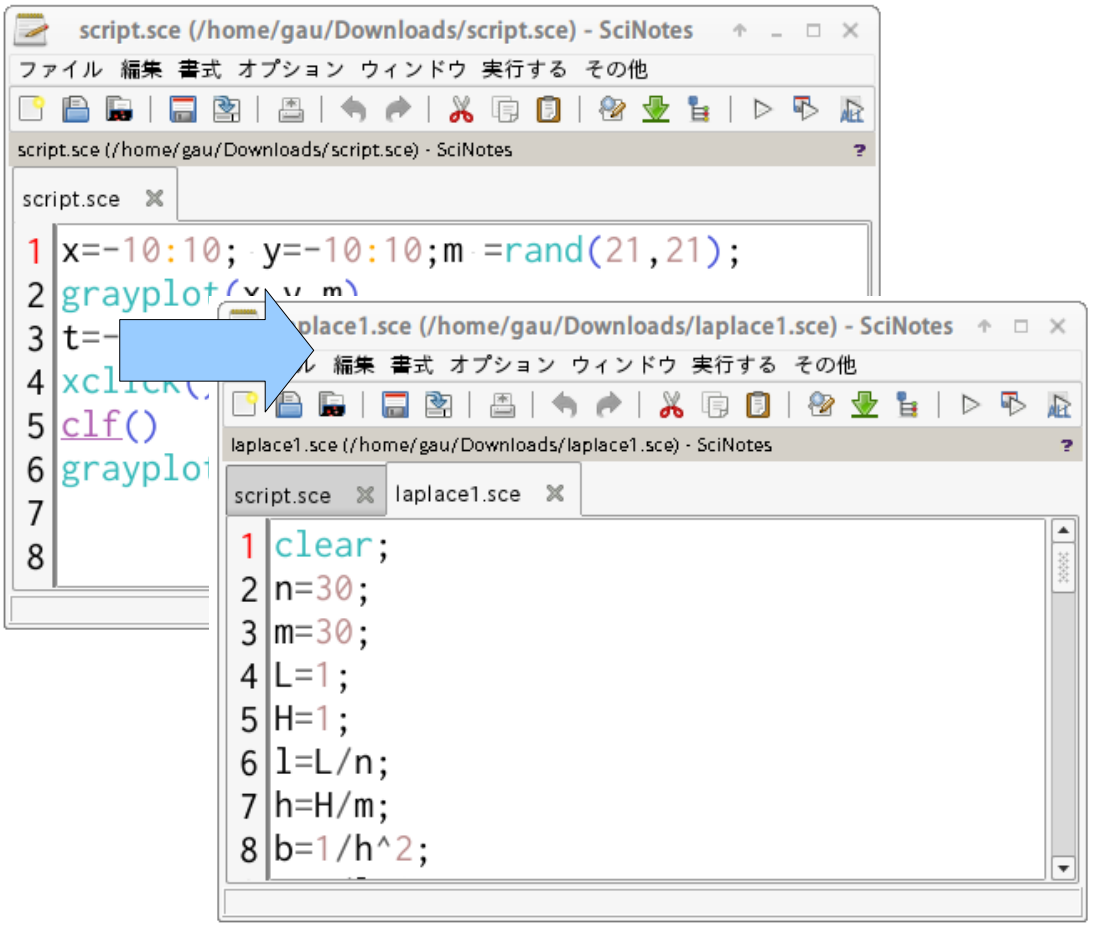
Scilabコンソールではなく、SciNotesからもスクリプトファイルを開くことができます。

開くボタン　を押すと「開く」ウィンドウが表示され、ファイルを選択できます。

開いたスクリプトは追加のタブに表示されます。

新規ボタン　で新しいスクリプトの編集を開始することもできます。

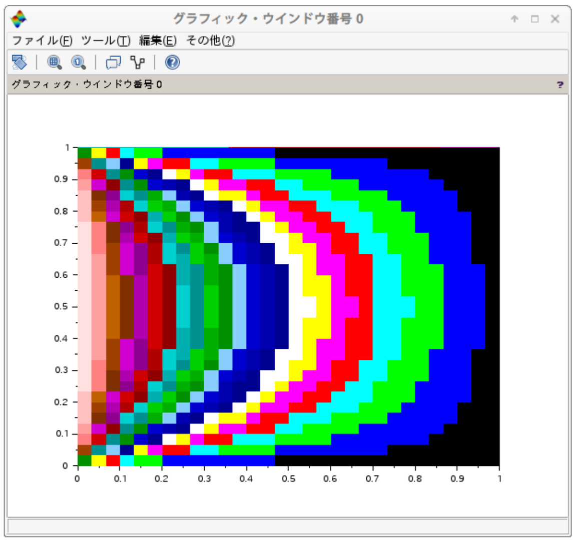
スクリプトの編集や実行の手順はscript.sceの場合と同様です。

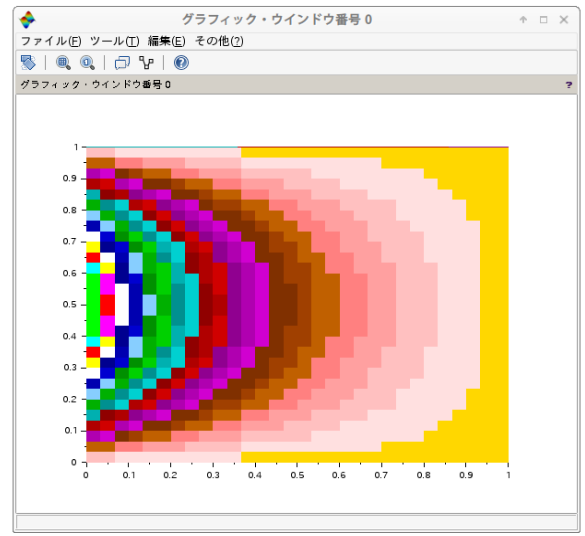


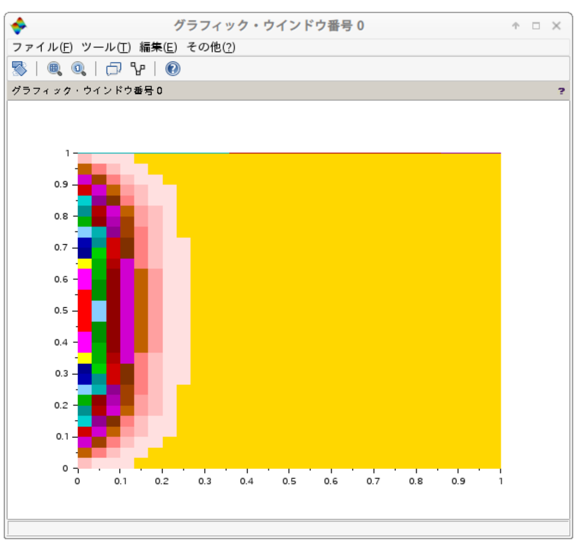
課題1

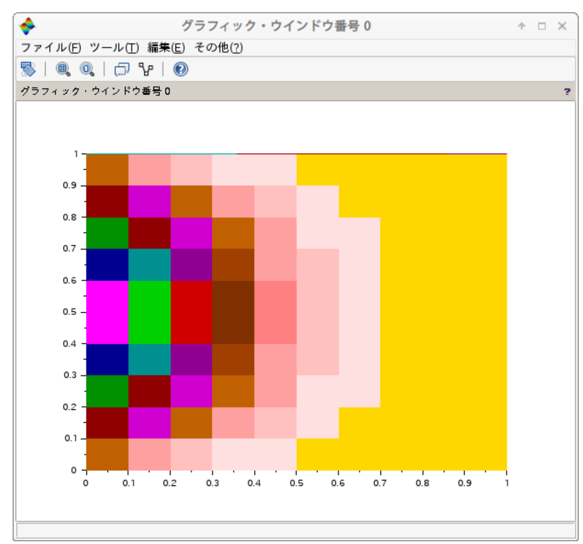
4つのサンプルスクリプトを実行しましょう。

Scilabコンソールからでも SciNotesからでも良いので、4つのスクリプトの読み込みと実行をしてみてください。









サンプルスクリプト解説

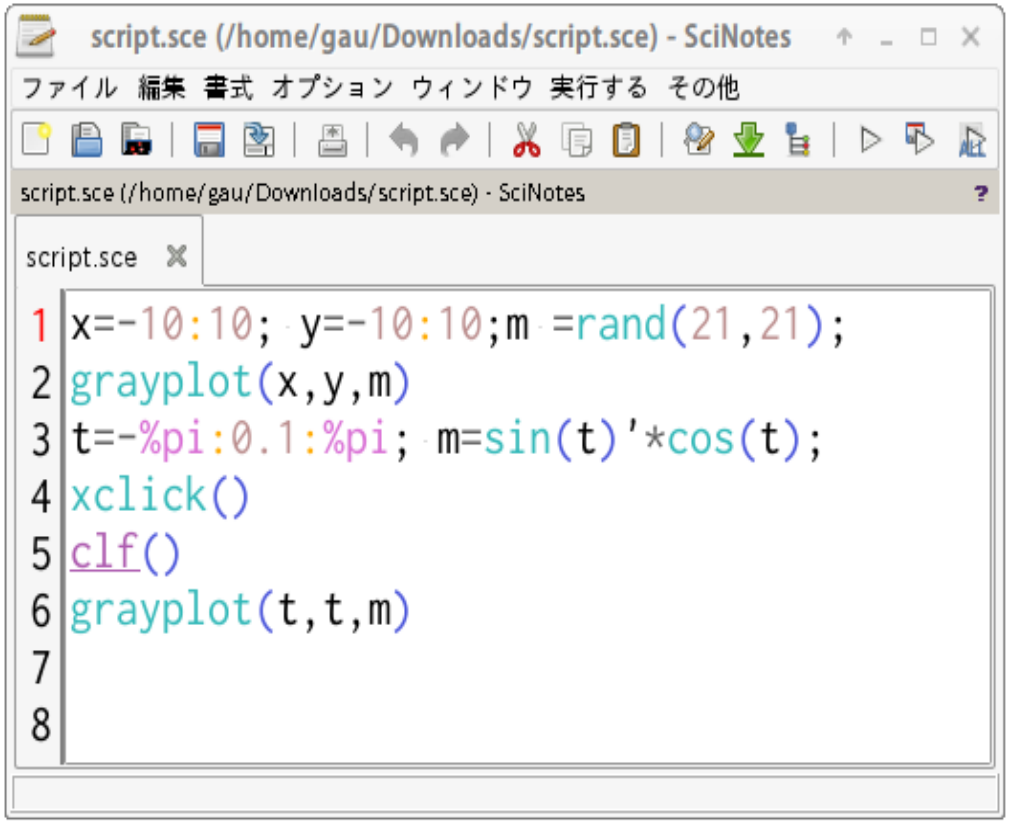
script.sceを解析しましょう。

sript.sceには繰り返しや条件分岐等の制御構造が含まれていません。

1行目から順に各行の命令が実行されるだけのものです。

実際に、各行をScilabコンソールに貼り付けて実行していけば、スクリプトを実行したのと同じ結果を得ることができます。

以降のスライドで行毎の内容について説明します。



サンプルスクリプト解説

1行目：

```
1 x=-10:10; y=-10:10;m =rand(21,21);
```

区切文字「;」（セミコロン）で区切られた 3つの代入命令です。

「;」は命令の返値表示を抑制するので、代入結果のコンソールへの表示はありません。

xへの代入の右辺では「:」による行列の生成が実施されます。

-10:10 は [-10:1:10] の略記と考えられ、次の行ベクトルを生成します。

[-10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4 5 6 7 8 9 10]

yへの代入の右辺も同一なので、説明は省略します。

mへの代入の右辺はrand関数による行列を生成します。

整数2つを「,」カンマで区切った場合はそれぞれ生成される行列の行数と列数に対応し、rand(21,21)の場合は21行21列の行列になります。

行列の成分は単一の乱数系列から作られた[0,1)の区間の乱数です。

乱数系列の詳細はhelpコマンドを用いて調べてください。

サンプルスクリプト解説

2行目：

```
2 grayplot(x,y,m)
```

grayplotは2次元平面の数値分布を色で表すカラープロットを作成します。

xは長さn1、yは長さn2のベクトル、mはn1×n2の実数行列を与えます。

mの成分m(j,k)は2次元平面の座標(x(j),y(k))における数値を与えます。

grayplotは座標（x(j),y(k)),(x(j+1),y(k)),(x(j+1,y(k+1)),(x(j),y(k+1))の4点で囲まれた長方形をm(j,k),m(j+1,k),m(j+1,k+1),m(j,k+1)の4つの数値の平均値に対応する色で塗り潰します。

値と色との対応はグラフィックウィンドウに設定されるカラーマップcolor_mapで決まります。設定されているcolor_mapは次の手順で確認できます。

h=gcf(); h.color_map

*.color_mapはn×3の実行列で各列がRGBの強度を表し、行毎に数値に対応する段階を表します。

通常の利用では *.color_map(1,*) がgrayplotのmにおける最小値に対応するRGBの強度を与えます。

サンプルスクリプト解説

2行目：

```
2 grayplot(x,y,m)
```

gcf()を利用して取り出したカラーマップは変更も可能です。

script.sceを実行した後のScilabコンソールで以下を実行すればcolor_mapに設定された行列が表示されます。

h=gcf(); h.color_map

h.color_mapに行列を代入すれば、グラフのカラーマップを変更することができます。例えば、次の命令を実行すればランダムなカラーマップになります。

h.color_map=rand(32,3);

適切なカラーマップ行列を作成するのは手間なので専用の関数が用意されています。

h.color_map=graycolormap(32)

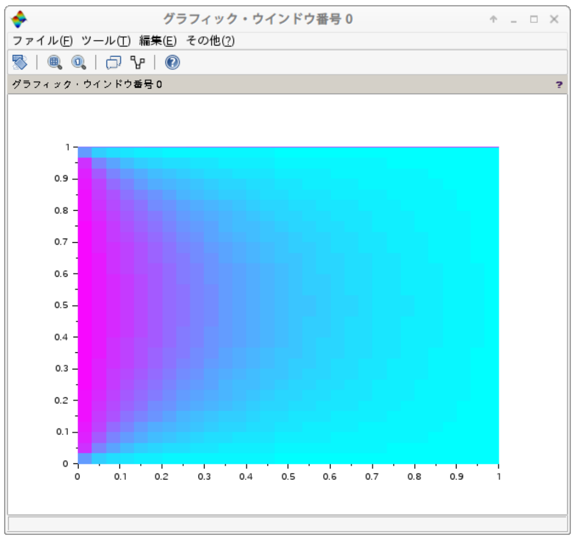
は黒→白の32段階のグラデーションから成るカラーマップ行列を代入します。同様に引数で段階数を指定できる関数には以下のようなものがあります。

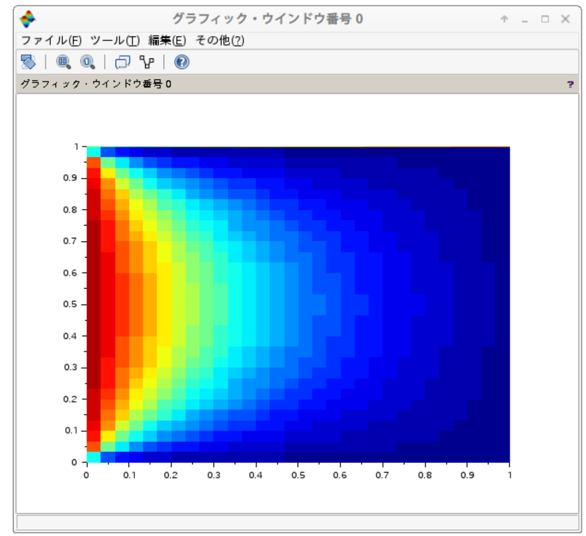
coolcolormap(),jetcolormap(),oceancolormap(),pinkcolormap() 他

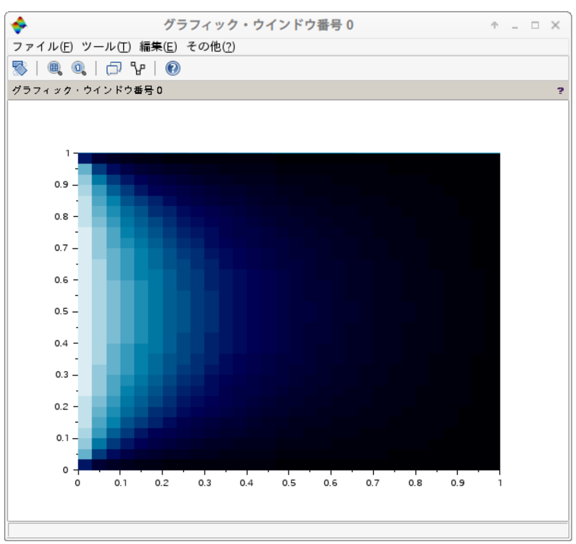
課題2

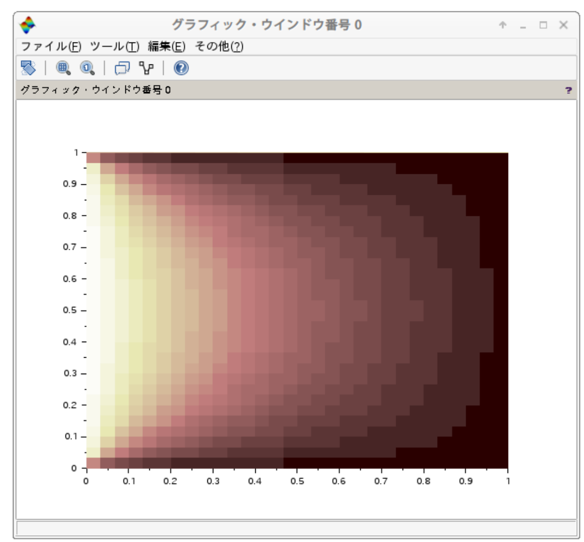
温度分布に適したカラーマップを見つけましょう。

Laplace方程式、熱伝導方程式のプログラムを実行した後のScilabコンソールでカラーマップを変更して温度分布らしいグラフになるものを見つけてください。









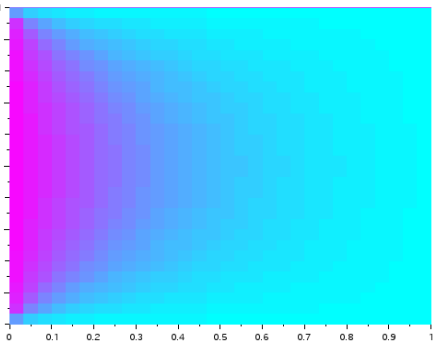
課題2

解答、ではありませんが。

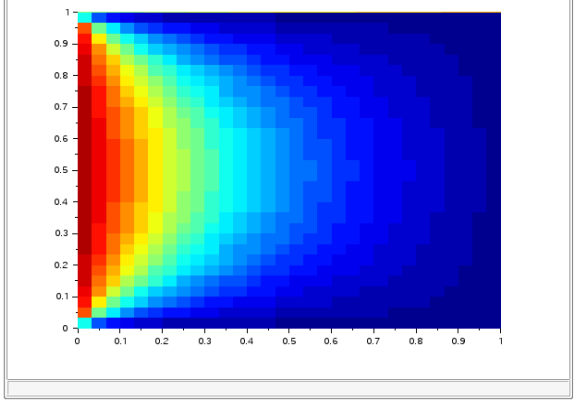
温度分布に適したカラーマップを見つけましょう。

Laplace方程式、熱伝導方程式のプログラムを実行した後のScilabコンソールでカラーマップを変更して温度分布らしいグラフになるものを見つけてください。

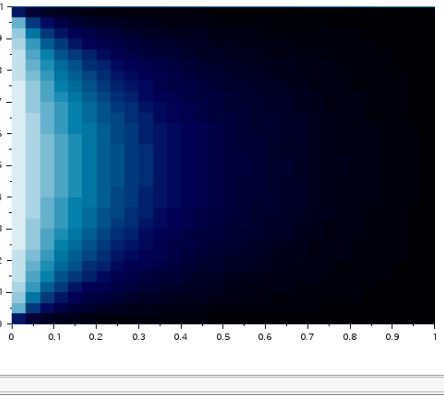
これはcoolcolormap()です
```
h=gcf(); h.color_map=coolcolormap(32);
```



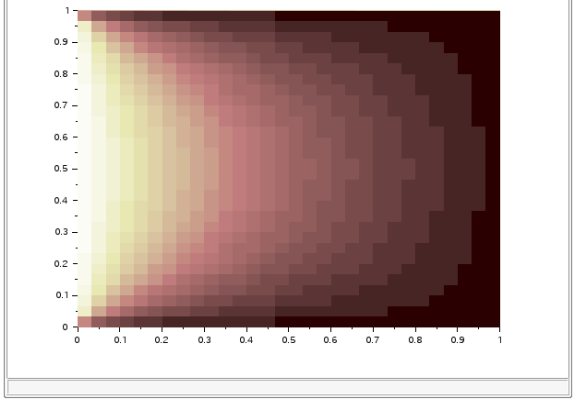
これはjetcolormap()です
```
h=gcf(); h.color_map=jetcolormap(32);
```



これはoceancolormap()です
```
h=gcf(); h.color_map=oceancolormap(32);
```



これはpinkcolormap()です
```
h=gcf(); h.color_map=pinkcolormap(32);
```



サンプルスクリプト解説

3行目：

```
3 t=-%pi:0.1:%pi; m=sin(t)'*cos(t);
```

2つの代入命令です。

tの右辺は第1成分 -%pi=−π、最終成分 %pi=πのベクトルを生成します。0.1は−πからπの間の増分です。[-%pi:0.1:%pi] としても同じ結果になります。

mの右辺はsin(t)’とcos(t)の積ですが、tがベクトルなのでsin(t)もcos(t)もtの各成分を引数とするsin(),cos()の返値を成分とするベクトルになります。

[-%pi:0.1:%pi]は行ベクトルになるので1行63列の行列と見做すこともできます。sin(t)もcos(t)も引数と同じ大きさの行列になります。

sin(t)’の「’」正確にはエルミート転置（転置して成分をその複素共役に置き換える）を表しますが、実行列なので通常の転置行列と考えても同じです。
（成分によらず転置行列を与える場合は「’」の代わりに「.’」を使います。）

sin(t)’は63行1列、cos(t)は1行63列の行列なので、その積は63行63列の行列になります。

サンプルスクリプト解説

4行目：

```
4 xclick()
```

最初にscript.sceに制御構文が含まれていないと説明しましたが、xclick()はマウス操作を待つので入力に対するブロック動作をします。

返値にクリックされたボタンや座標、ウィンドウやメニューについての情報も含まれますが、script.sceでは全く利用していないので、単にマウスクリックされるまで処理を停止するためだけのものになっています。

詳細はhelp xclickとしてください。

5行目：

```
5 clf()
```

グラフィックのクリアとリセットをします。

引数を指定することで、特定のグラフィックウィンドウやさらにその一部分等を指定してクリアやリセットをすることもできますが、script.sceでは単一のグラフィックウィンドウについて、表示内容を消去するためだけに使われています。

詳細はhelp clfとしてください。

サンプルスクリプト解説

6行目：

```
6 grayplot(t,t,m)
```

grayplot()は既出です。

tは−π〜πの目盛、mはsin(x)cos(y)の数値が入っているので、この2次元グラフをカラーマップで表現したものが表示されるはずです。

レポート(11)

学籍番号・氏名を記し提出してください。

- Scilabによる差分解法の演習を実施して、4つのサンプルスクリプトの問題と解法を説明してください。

授業レポート用紙:氏名(　　)学籍番号(　　)

2020年1月20日（月）

できれば授業の感想も書いてください。