

情報工学演習 D
ガイダンス資料

室蘭工業大学
情報工学科

2009 年 10 月

1 はじめに

実社会においては、まず社会的な要求を想定し、それを満たすために目的の設定が行われ、目的達成のための手段が考えられた後、成果物(物やプログラムなど)が作られる。作られた後も様々な評価が行われ、改良等に役立てられ、時に新しい社会的要求・目的となる。例えば車であれば、まず社会的な要求として『乗り心地の良い車』が想定される。これを満たすための目的としては、サスペンションの改良・エンジンなどの騒音の低減化・ハンドルやシートなどのさわりの心地や座り心地など、様々なものが考えられる。一般には複数ある目的から一つを選択し、それについて実現が考えられる。例えば上記の目的の中からシートの座り心地の向上に注目すると、そのための手段としては材質を選択したりシートの形状を考慮して包み込み感を向上することなどが考えられる。これらの結果を数値化し、客観的に比較・検討を行うことで、よりよいものの作成が行われる。更にもしかすると、これらの改良・改善の結果、ただ座り心地がよいものよりも、マッサージ機能や眠り防止機能など刺激のあるシートが欲しいという要求がうまれてくるかも知れない。その場合、新たな目的として開発・研究が行われるだろう。

このように実社会において求められる能力として、下記のものが考えられる。

- 目的を設定する能力
時代背景にあわせて適切な目的を発見し、設定できる能力。
- 目的達成のための手段を選択できる能力
目的を実現するために手段を選択できる能力。更には目的達成のために新たな知識を学習し、それを新たな手段として選択できる能力。
- 手段を用いて結果を出し適切に考察する能力
手段を用いることで様々な結果を得られるが、得られた結果から必要な情報を切り出し、分かりやすく提示できる能力。また得られた結果から、客観的に納得してもらえ考察を導き出す能力。
- 文書化し知識の継承を行う能力
自身が行った諸々を文書化し他者へ伝える能力。主に「文章作成能力」と「図形作成能力」。一般に行った研究・実験・業務は、いつでも・同じように他者へ伝えるために文書化する必要がある(口頭での伝承は記憶の劣化が簡単に起こる)。

2 本演習のねらい

本演習では、単なる知識(プログラミングスキル)を向上させるためだけの演習ではなく、知識を用いて目的達成能力を向上させるための演習と考える。特に「1 はじめに」で述べたように、目的・手段・結果・考察について焦点を当て、その流れを学習することを目的として行う。

2.1 演習上の目的:人口推移シミュレーション

具体的な目的の一つとして、個体数推移シミュレーションの実現を考える。シミュレーションとは、模擬実験である「模擬」の意味どおり、本当に実験を行おうとすると費用がかかりすぎたり実験の規模が大きすぎて実現できないため、行いたい実験を、行える範囲で真似て実験を行うもので

ある．例えば，街の中に高層ビルを建てた時の風の流れとその影響を知りたい場合，実際に高層ビルを建てて実験を行うのは現実的ではない．そこで，1/10 スケールのミニチュアを作成し，模擬実験が行われたりする（風洞実験）．その他にも，現在はゲームとして知られている将棋・チェスなども，軍事シミュレーション（戦争の模擬実験）として発達したと考えられている．

このように，従来では代替物（ビルに対してミニチュア，兵に対して駒など）を使って模擬実験を行っていたものが，現在ではコンピュータを用いた模擬実験が盛んになってきている．その理由は，コンピューティング能力の飛躍的な向上により，複雑な物理計算を瞬時に行うことが可能になったことが挙げられる．これにより現実空間をコンピュータ上に高い精度で再構成することが可能となり，プログラム次第で様々な実験を行うことが可能になった．ただし，現実と全く同じ状況を作ることは労力的に不可能である．対象とする実験に関する要因の中で必要なものと不必要なものを切り分け，プログラム可能なくらい簡単化する必要が出てくる．これをモデル化という．このモデル化によって実験を簡単化して対象の模擬実験を行うことを特にコンピュータシミュレーションという．本演習では人口や動植物の個体数の推移に対するシミュレータ（模擬実験を行うプログラム）の作成を目的とする．

2.2 演習上の目的:バンディット問題を解くプログラムの作成

もうひとつの目的として，バンディット問題を解くプログラムの作成を考える．バンディット問題とは，簡単に言えばゲームである．ゲームであるために，そこにはルールが存在し，それに則って勝敗等が決められる．ここでゲームの相手はバンディット・マシンというスロットマシンのようなものである．ルールは存在するが個々のマシンがどのように動くのかは分からない．分からない相手に対してどのように勝つかを考え，勝てるプログラムの作成という形で具現化/具体化することが目的である．人間の試行錯誤による常勝の道の発見と，それを明確にルール/アルゴリズム化することを学んで欲しい．

3 演習の進め方

演習は，1つの対象を複数週間かけて行う．対象は，レポートの形式で配布する．レポートは以下の内容を含んでおり，足りない部分を各人が書き足すことで自身のレポートを作成する．

1. はじめに
2. 目的・製作対象
3. アプローチ
4. 手法
5. 実験
6. 結果
7. 考察
8. 参考文献

この中で主に各人が作成する部分は、6 結果および 7 考察、参考にした本や HP があれば、8 参考文献である。

その他の 1 から 5 までは、ほぼ書いてあるので、それを参考にしながら演習を行う。ただし、最初から用意してある文章だけだと不足である、と感じた場合は書き足すこと。同様に、用意してある文章・部分が不要である、と感じた場合は削除すること。

4 一般論:レポートについて

4.1 質と量

レポートを含む全ての文書は、読み手の為に存在する。書き手がいかにすばらしいことを考えたり行ったりしたとしても読み手に伝わらなければ完全に無意味である。故に、書き手は量を書くことよりも書き手に自身の伝えたいことを十分に伝えられるよう質を高める努力をしなければいけない。その一方で、質を客観的に評価することは難しく、そのため質を高める努力をいかにするか、ということは難しい。ここで質を高めるための一つの方法として、量を書く、ということがある。それは質の低下を招く要因が量の不足があるためである。一般に人間は自分の持つあたりまえの常識・知識は他人も持つと考えがちである。文章を書く際にも、自分にとって当たり前のことは省きがちである。加えて、当たり前すぎることは最初から自分の意識にのぼらないことが多い。しかし、この「自分のもつ」当たり前は他人にとっても「当たり前」は思ったほどなりたない。よって、自分にとって当たり前のことを省いてある文章は、他人には読みづらいことになる。そこで読みやすい文章を書くための最初のステップは、「自分にとって当たり前と思うことでも、懇切丁寧に文章にしていく」ことであり、その評価の一つが文章量となる。

4.2 誰が為の名前

学校のみならず、社会において文書を書くことは多々ある。その際に、「誰の報告書であるか」を示す名前を必ず記述する。それは単に報告書を書いた人を現すだけでなく、往々にして書いた内容について責任を持つ人の意味を持つ。最低限度の責任として、他の書籍等を引用・参考にしようが自身で言葉を作り出そうが、報告書に書いたことの責任は報告者に帰する。つまり、虚偽・間違いなどが在った場合には、報告者自身が責任をとらなければいけない。そのため、報告書の内容は報告者がきちんと把握していなければならず、報告者が書いた以上、「他の文献をコピーしただけです」という言い訳は成り立たない。

一方、そのように責任を負うので、自身の書いたものについて自由に使用できる権利を持っている。つまり、乱暴にいうと「この文章は自身が完全に責任を負う。だから自由に使用できるのも自分のみなのだ」ということである。他の文章を引用する場合には、その文章を書いた人間の苦勞等を考慮に入れた上で用いる必要がある。

他の人と相談することや他の書籍・HP の情報などを参考にすることは大いに薦める。人間は一人よりも多人数で事を運ぶ方が効率がよいし、高度なことを成すことができる。しかし、他の人の情報を、理解なしに丸ごとコピーすることは他人にとっても自分にとっても不幸なことであるのでしてはいけない。

A TeX の使い方

本演習では TeX を使用して配布資料の閲覧・レポートの作成を行う。特に、配布資料への加筆という形でレポートを作成するため、TeX の使用が必須となる。以下に TeX を用いた文書作成手順を簡単に述べる。詳しくは、[1, 2] やインターネットで使用方法を検索して用いること。

A.1 資料の準備

まず本演習では資料の準備から始める。資料ははたおり虫の所定の位置にあり、tar 圧縮してある。よってまず、ダウンロードし解凍を行う。その後 TeX にてコンパイルし資料とする。その中に、実験の目的等があるのでそれを見ながら実験を行う。実験を行った後、資料の該当部分を加筆し、また不必要と思われる部分は削除し、報告書として完成させて提出する。

1. tar ファイルのダウンロード
はたおり虫の所定の場所よりダウンロード
2. tar ファイルの解凍
tar xvf hoge.tar (ダウンロードファイル名の例:hoge.tar)
3. TeX ファイルのコンパイル・確認
platex hoge.tex (TeX ファイル名の例:hoge.tex)
platex hoge.tex (2 回同じコマンドを実行)
xdvi hoge.dvi& (閲覧)
4. TeX ファイルの修正
実験等による加筆など
hoge.tex を修正・加筆
5. 印刷
3 から 4 を繰り返し十分なものが出来上がったから印刷
dvi2pdf hoge.dvi(pdf ファイルに変換:hoge.pdf の作成)
open hoge.pdf (pdf ファイルをアプリケーションで開く)
アプリケーションを利用して印刷

A.2 図・グラフの作成

図、特にグラフは gnuplot を用いて作成する。gnuplot はコマンドまたはファイルからデータ等を読み取り、グラフ化するソフトである。

A.2.1 起動と終了

起動するには、コマンドプロンプト (%) で以下のように入力する。
% gnuplot
また終了するには gnuplot の中で
% quit
を入力する。

A.2.2 ファイルからグラフ化

ファイルからデータを読み込んでグラフ化する。ファイルの書式は、空白（スペースやタブ）で区切られた数値データとなる。区切り文字により、一列目のデータ、二列目のデータとなる。一般に一列目のデータが x 軸の値となり、二列目のデータが y 軸の値となる。

以下に gnuplot の中で使用できる代表的なコマンドを記載する。詳しくは、[3]-[6]などを参考にすること。

- plot "ファイル名"
ファイル名にあるデータを自動的に解釈してグラフ化
- plot "ファイル名" notitle
データの種類の名前を非表示
- plot "ファイル名" with A
A:points データを点で表示
A:lines データを結んで線にする
A:linespoints 線にした上で点も打つ
- plot "ファイル名" using a:b
ファイル名の a 列目を x 軸、b 列目を y 軸としてグラフ化
例:plot "hoge.txt" using 1:2
- plot "ファイル名" using a:(f(\$b))
ファイル名の a 列目を x 軸、b 列目を関数 f により演算してグラフ化。関数を使用して演算する場合は、 $\$b$ と列を表す数字の前に $\$$ をつける。
例:plot "hote.txt" using 1:(log10(\$2))

A.2.3 ファイルから色々な設定

gnuplot では、表示領域の設定など様々な設定をコマンド set で行うことができる。しかし、毎回入力するのは面倒である。そこで設定したい項目をファイルに記述することで、そのファイルから設定を読み込むことができる。

```
% load "ファイル名"  
ファイル名より設定を読み込む。
```

A.2.4 複数のグラフをひとつの図に

コマンド plot は後に続くファイル・式をグラフ化するものである。これによってグラフ化すると前のグラフは消えてしまう。

```
% plot sin(x)  
% plot cos(x) ← sin(x) が消されて cos(x) のグラフのみになる。  
replot を使用すると、前のグラフに追加する形で新しいグラフを追加することができる。  
% plot "データ"  
% replot cos(x) ← "データ"ファイルのグラフに加えて cos(x) のグラフが描画される。  
よって、replot のみで使うと、直前に表示したグラフが再描画される。
```

```
% plot "データ"  
% replot "データ"ファイルのグラフを再描画
```

A.2.5 EPS 化 , および TeX への貼り付け

作成したファイルは EPS ファイルで保存することができる . このファイルを TeX に埋め込むことで図を文章中に埋め込むことができる .

- plot "hoge.txt"... ← EPS ファイル化したいグラフを表示し最終確認
- set output "hoge.eps" ← ファイル名のセット
- set term postscript eps "フォント名" フォントの大きさ (数字で) ← 出力先を画面からファイルへ . なおフォント名より後ろは省略可能 .
- replot ← (ファイルへ) 再描画
- set term x11 ← 出力先を画面に戻す .

一方作成した EPS ファイルを TeX 内に書き込むためには , TeX ファイルの先頭付近に

```
\usepackage[dvips]{graphicx}
```

を書く . またグラフを挿入したい場所に ,

```
\begin{figure}[htb]  
\includegraphics[scale=.7]{hoge.eps}  
\caption{図の題名 \label{fig:hoge.eps}}  
\end{figure}
```

と記入する .

作成した図は , kterm 上で以下のコマンドを使用することで表示できる .

- gv hoge.eps

A.3 表の作成

表は TeX のコマンドとして作成することができる . 図 1 で表 1 が作成できる .

表 1: 表の書式テスト

テスト	テスト	テスト
とてもテスト	すごいテスト	めちゃめちゃテスト

```

\begin{table}[hbt]
\begin{center}
\caption{表の書式テスト \label{tb:pattern}}
\begin{tabular}{|c|l|r|}\hline
テスト & テスト & テスト \\ \hline
とてもテスト & すごいテスト & めちゃめちゃテスト \\ \hline
\end{tabular}
\end{center}
\end{table}

```

図 1: 表の書式テスト

B 参考文献

参考文献

- [1] 生田 誠三, "LATEX 文典", 朝倉書店, 1996
- [2] 海上 忍, 黒川 弘章, "これだけでできる LATEX 実践活用ガイド", 技術評論社, 2000
- [3] <http://auemath.aichi-edu.ac.jp/khotta/ghost/gnuplot.html>
- [4] <http://t16web.lanl.gov/Kawano/gnuplot/>
- [5] <http://tortoise1.math.ryukoku.ac.jp/takataka/gnuplot/>
- [6] <http://www.al.cs.kobe-u.ac.jp/inamoto/unix-tools/useful/gnuplot/>