

# プログラミング 第一回 レポート

202212022 田島瑞起

2023/05/17

## 1 はじめに

今回の課題では、

1. Linux のファイル操作コマンドを表形式でまとめる。
2. C 言語プログラムを図（テキスト形式）として掲載する
3. Linux コマンドを操作しているところを図として掲載する
4. 本の紹介をする 以上 4 点の設問に対し LaTeX を用いて解答する。

以上四点の設問を満たす pdf ファイルを LaTeX を用いて作成する。

## 2 Linux 操作コマンドについてまとめる

### 2.1 課題内容の説明

課題内容はこの講義で取り扱ったファイル操作に関するコマンドを表形式で纏めることである。具体的な要件として、下記五つを満たす必要がある。

1. 操作オプションの異なるものは別の行として考える
2. 一行はコマンド名、コマンド例、説明を列とする
3. 7 12 行の範囲に収まるようにコマンドを選ぶ
4. 節の中には、本文を記述して、表番号を参照するようにすること
5. PDF 化する際、表がページからはみ出ないように調節すること

## 2.2 課題への取り組み方針

Linux コマンドにて最も重要であるディレクトリ操作一般について纏める。

## 2.3 解答結果

表 1 Linux コマンド

コマンド名	コマンド例	コマンド説明
ls	ls	カレントディレクトリ配下を表示
ls -a	ls -a	隠しファイル含めすべてを表示する
pwd	pwd	現在のパスを表示する
cd 元 1	cd test	対象ディレクトリに移動する
cd ../	cd ../	親ディレクトリに移動する
mkdir 元 1	mkdir test	対象ディレクトリを作成
rm 元 1	rm test.tex	対象ファイルを削除する
rmdir 元 1	rmdir test	対象ディレクトリを削除する

## 2.4 解答結果に対する説明

table 環境を使用して表を作成し、題意を満たすように列幅を変更して文字がはみ出ないように調節した。(表 1)

## 2.5 考察

(表 1) は上述された 5 つの要件を満たしている。表は一行ごとに色が変わると見やすいと感じたので次回以降実装する。

## 3 プログラムを図として掲示せよ

### 3.1 課題内容の説明

C 言語のソースプログラムを図として挿入し PDF 化する。PDF 化する際に下記要件を満たす必要がある。

1. プログラムの各行に番号をつけること
2. プログラムの各文字をテキストとしてあるかえること
3. プログラム部分はタイプライタ体で表示されること
4. 本文を作成して図を参照すること
5. PDF 化した際にページをはみ出ないこと

### 3.2 課題への取り組み方針

`cat -n` コマンドを用いてソースコードに行番号を付与した後に、そのテキストを、`verbatim` 環境を用いてそのまま埋め込む。

### 3.3 解答結果

```
1      1  #include <stdio.h>
2      2
3      3  void sort(int n, int num[]){
4      4      int i, j;
```

```

5      5
6      6      for(i=0; i<n-1; i++){
7      7          for(j=i+1; j<n; j++){
8      8              if (num[i] > num[j]){
9      9                  int tmp;
10     10                  tmp = num[i];
11     11                  num[i] = num[j];
12     12                  num[j] = tmp;
13     13              }
14     14          }
15     15      }
16     16  }
17     17
18     18  int main(int ac, char* av[]){
19     19      int num[] = { 3, 1, 4, 5, 9, 2, 6, 8, 7 };
20     20      int i;
21     21
22     22      for(i = 0; i < 9; i++){
23     23          printf("%3d", num[i]);
24     24      }
25     25      printf("\n");
26     26
27     27      sort(9, num);
28     28
29     29      for(i = 0; i < 9; i++){
30     30          printf("%3d", num[i]);
31     31      }

```

```
32      32      printf("\n");  
33      33  }
```

### 3.4 解答結果に対する説明

課題の取り組み方針にて書いたとおりに実行した

### 3.5 考察

ソースコード表は、上述した要件 5 つを満たしている。ソースコードも行ごとに色が変わると見やすいと感じた。

## 4 Linux 操作中の画面を図として挿入する

### 4.1 課題内容の説明

全額計算機システムの LINUX 環境を使用しているところをキャプチャして、図として挿入する。今課題は下記要件を満たす必要がある。

1. 何をしているところなのか説明を入れること
2. どのキャプチャ方法を用いたのか示すこと
3. そのキャプチャ方法の操作方法を説明として含めること
4. 図を Latex の機能で参照すること
5. PDF 化した際にページをはみ出ないこと

## 4.2 課題の取り組み方針

今回は WindowsPc を用いてスクリーンショットを取り、それを figure 環境にて埋め込む。

## 4.3 解答結果

```
(/usr/local/texlive/2022/texmf-dist/tex/latex/etoolbox/etoolbox.sty)
(/usr/local/texlive/2022/texmf-dist/tex/latex/jlreq/jlreq-helpers.sty)
jlreq guessed engine: lualatex
(/usr/local/texlive/2022/texmf-dist/tex/luatex/luatexja/luatexja.sty
(/usr/local/texlive/2022/texmf-dist/tex/luatex/luatexja/luatexja-core.sty
(/usr/local/texlive/2022/texmf-dist/tex/luatex/luatexbase/luatexbase.sty
(/usr/local/texlive/2022/texmf-dist/tex/luatex/ctablestack/ctablestack.sty))
(/usr/local/texlive/2022/texmf-dist/tex/generic/ltxcmds/ltxcmds.sty)
(/usr/local/texlive/2022/texmf-dist/tex/generic/pdfTeXcmds/pdfTeXcmds.sty
(/usr/local/texlive/2022/texmf-dist/tex/generic/infwarerr/infwarerr.sty)
(/usr/local/texlive/2022/texmf-dist/tex/generic/iftex/iftex.sty))
(/usr/local/texlive/2022/texmf-dist/tex/latex/xkeyval/xkeyval.sty
(/usr/local/texlive/2022/texmf-dist/tex/generic/xkeyval/xkeyval.tex
(/usr/local/texlive/2022/texmf-dist/tex/generic/xkeyval/xkutils.tex
(/usr/local/texlive/2022/texmf-dist/tex/generic/xkeyval/keyval.tex)))
(/usr/local/texlive/2022/texmf-dist/tex/latex/everyhook/everyhook.sty
(/usr/local/texlive/2022/texmf-dist/tex/latex/svn-prov/svn-prov.sty))
(/usr/local/texlive/2022/texmf-dist/tex/luatex/luatexja/ltj-base.sty)
(/usr/local/texlive/2022/texmf-dist/tex/luatex/luatexja/ltj-latex.sty
(/usr/local/texlive/2022/texmf-dist/tex/luatex/luatexja/patches/ltjfont.sty
(/usr/local/texlive/2022/texmf-dist/tex/latex/base/tuenc.def))
(/usr/local/texlive/2022/texmf-dist/tex/luatex/luatexja/patches/ltjdefs.sty
(/usr/local/texlive/2022/texmf-dist/tex/luatex/jlreq/jfm-jlreqv.lua)
(/usr/local/texlive/2022/texmf-dist/tex/luatex/jlreq/jfm-jlreq.lua))
(/usr/local/texlive/2022/texmf-dist/tex/luatex/luatexja/patches/ltjcore.sty)
(/usr/local/texlive/2022/texmf-dist/tex/luatex/luatexja/patches/ltj-atbegshi.
sty)
(/usr/local/texlive/2022/texmf-dist/tex/luatex/luatexja/patches/ltj-geometry.
sty (/usr/local/texlive/2022/texmf-dist/tex/generic/iftex/ifluatex.sty)))
(/usr/local/texlive/2022/texmf-dist/tex/luatex/luatexja/luatexja-compat.sty)
(/usr/local/texlive/2022/texmf-dist/tex/luatex/luatexja/addons/luatexja-adjust.
sty) (/usr/local/texlive/2022/texmf-dist/tex/latex/lm/lmodern.sty)
(/usr/local/texlive/2022/texmf-dist/tex/latex/jlreq/jlreq-complements.sty
(/usr/local/texlive/2022/texmf-dist/tex/latex/filehook/filehook.sty
(/usr/local/texlive/2022/texmf-dist/tex/latex/filehook/filehook-2020.sty)))
(/s2212022.aux) (/usr/local/texlive/2022/texmf-dist/tex/latex/base/tslcmr.fd)
(/usr/local/texlive/2022/texmf-dist/tex/latex/lm/otlrmr.fd)
(/usr/local/texlive/2022/texmf-dist/tex/latex/lm/omllmm.fd)
(/usr/local/texlive/2022/texmf-dist/tex/latex/lm/omslmsy.fd)
(/usr/local/texlive/2022/texmf-dist/tex/latex/lm/omxlmex.fd)
! I can't write on file `s2212022.pdf'.
Please type another file name for output: s2212022_17.pdf
[1{/usr/local/texlive/2022/texmf-var/fonts/map/pdfTeX/updmap/pdfTeX.map}]
Overfull \hbox (30.79128pt too wide) in paragraph at lines 32--44
[1][1][1]
[2] [3] [4] (./s2212022.aux)
(see the transcript file for additional information)
889 words of node memory still in use:
8 hlist, 1 vlist, 5 rule, 2 glue, 3 kern, 2 glyph, 164 attribute, 47 glue_sp
ec, 24 attribute_list, 1 write, 10 user_defined_nodes
avail lists: 1:5,2:24106,3:633,4:273,5:64,6:77,7:2307,8:1,9:1169,10:15,11:12
</usr/local/texlive/2022/texmf-dist/fonts/opentype/public/lm/lmmonol2-regular.o
tf></usr/local/texlive/2022/texmf-dist/fonts/opentype/public/lm/lmroman10-regul
ar.otf></usr/local/texlive/2022/texmf-dist/fonts/opentype/public/haranoaji/Hara
noAjiGothic-Medium.otf></usr/local/texlive/2022/texmf-dist/fonts/opentype/publi
c/lm/lmsans10-bold.otf></usr/local/texlive/2022/texmf-dist/fonts/opentype/publi
c/lm/lmroman12-regular.otf></usr/local/texlive/2022/texmf-dist/fonts/opentype/p
ublic/lm/lmroman17-regular.otf></usr/local/texlive/2022/texmf-dist/fonts/openty
pe/public/haranoaji/HaranoAjiMincho-Regular.otf>
Output written on s2212022_17.pdf (4 pages, 74990 bytes).
Transcript written on s2212022.log.
s2212022@icho01:~/Programing2023/report1$
```

図1 linux を操作している画面

## 4.4 解答に対しての説明

- 1 を取り組み方針の通り解答した

## 4.5 考察

解答は上述した要件を満たしている。

# 5 本の紹介

## 5.1 課題内容

今課題では本の紹介をする。下記要件を満たす必要がある。

1. 本を三冊紹介する
2. そのタイトルの後ろに文献番号を参照する
3. BiBTeX を用いて文献情報を明示すること

## 5.2 課題の取り組み方針

好きな国内哲学者二人と、フーコーについて簡単に取り上げる。

## 5.3 解答結果

現代思想入門 [3] は現代思想というポストモダン思想について書かれた入門書であり、新書ということもあってか幅広くの人に読まれている。ポストモダン思想というのは、その思想自体を揶揄するようなワードとして一部使用されているが、簡単に要約すれば、白黒物事を決めたときに零れ落ちてしまうリアリティーについて取り扱う学問体系のことを指す。千葉はこの著作の中で入門書を読み終えた後にぜひ読んでほし

いという本を2つ紹介している。一つ目が「ジャックデリダについて」[1]である。この本はポストモダン思想家の中でも特別に重要なデリダについて、紹介した本である。デリダは際について着目した人間で、私たちが以下に二項対立から逃れられず、またそれが無意味であるということを鮮やかに導いた。次に紹介されているのが、「監獄の誕生」[2]である。フーコーは現在直面している私達の問題が歴史的に構成されているという事実を鮮やかに書き出した人物であり、この「監獄の誕生」では精神病者が監獄というモデルから生み出された産物であるという一見関連しないような対象の関係性を見事に見抜いた。以上3つの本は現代思想を国内で学ぼうとした時に有用な書であるから、是非を読んでほしい。

## 5.4 解答結果に対する説明

bib ファイルを生成して、`bibtex` コマンドを用いて、`bbl` ファイルを生成したのちに、本文中に番号付きで文献参照を行った。

## 5.5 考察

上述した要件を満たしている。文献参照は実装に必要なコマンドの順番を間違えるとエラーが起こり、また `lualatex` コマンドも二回打たないと実装できない点が面倒だった。

## 6 課題全体の感想

一回 `tex` ファイルをまとめて削除したことにより鬱状態に陥ったが、二回分演習したおかげで、何も見ずに `latex` を使いこなせるようになってよかった。重要なファイルは小まめに保存して、コピーを作成することの重要性に気が付いた。



## 参考文献

- [1] Hiroki Azuma. **存在論的、郵便的-ジャックデリダについて**. 新潮社, 1998/10/11.
- [2] Michel Foucault. **監獄の誕生**. 新潮社, 1977/09/22.
- [3] 千葉雅也. **現代思想入門**. 講談社, 2022/03/22.