



情報基礎ゼミナール

— 2001年度版 —

1年1組

福岡工業大学
情報工学部 情報工学科

柴田望洋

BohYoh Shibata

Fukuoka Institute of Technology

本資料について

- ◆ 本資料は、2001 年度・福岡工業大学 情報工学部 情報工学科 1 年生の講義

『情報基礎ゼミナール』

の補助テキストとして、福岡工業大学 情報工学部 情報工学科 柴田望洋が編んだものである。

- ◆ 参考文献・引用文献等は、資料の最後にまとめて示す。

- ◆ 諸君が本資料をファイルに綴じやすいように、研究室の学生達（卒研生と大学院生）が時間を割いて、わざわざ穴を開けるといいう作業を行っている（一度のパンチで開けることのできる枚数は限られており、気の遠くなるような時間がかかっている）。

必ず B 5 のバインダーを用意して、きちんと綴じていただきたい。

- ◆ 本資料のプログラムを含むすべての内容は、著作権法上の保護を受けており、著作権者である柴田望洋の許諾を得ることなく、無断で複写・複製をすることは禁じられている。

本資料は、Microsoft 社のワープロソフトウェアである Microsoft Word 2000 を用いて作成した。

本講義の成績評価について

本講義は、出席状況・レポート・ミニテスト等、さらに期末試験（内容は簡単なものとする予定）によって、成績評価を行う。欠席や遅刻をしないよう、留意されたい。

自己紹介

柴田望洋 (しばたぼうよう)

略歴

1963年11月 福岡県に生まれる
 1991年3月 九州大学大学院 博士後期課程 修了 (工学研究科 化学機械工学専攻) 工学博士
 同年4月 九州大学 工学部 化学機械工学科 助手
 1993年10月 国立特殊教育総合研究所 教育工学研究部 教育工学研究室 研究員
 1994年5月 福岡工業大学 工学部 情報工学科 助教授
 1995年4月 福岡工業大学 大学院 工学研究科 管理工学専攻 助教授 (兼任)
 1997年4月 福岡工業大学 情報工学部 情報工学科 助教授
 福岡工業大学 大学院 工学研究科 情報工学専攻 助教授 (兼任)

化学工学会、計測自動制御学会、日本教育工学会、日本工学教育協会、日本特殊教育学会、電子情報通信学会の会員

2000年 日本工学教育協会より著作賞を受賞

福岡陳氏太極拳研究会 事務局長

(1990-1992・全日本武術太極拳選手権大会・男子陳式の部 優勝)

足医師普及協力者

伝統的な中国秘術を指導しています。
 生徒募集中。気軽に問い合わせてください。

著書

| | |
|------------------|---|
| 秘伝 C言語問答 ポインタ編 | ソフトバンク, 1990 (韓国語版 KANAM PRESS, 1992) |
| C : 98 スーパーライブラリ | ソフトバンク, 1991 |
| CプログラマのためのC++入門 | ソフトバンク, 1992 |
| Dr.望洋のプログラミング道場 | ソフトバンク, 1993 |
| 明解 C言語 入門 | ソフトバンク, 1995 |
| C言語の鍛錬 | ソフトバンク, 1996 ★本学学生との共著★ |
| プログラミング講義C++ | ソフトバンク, 1996 (中国語版 博碩文化, 2000) |
| C++への道 | ソフトバンク, 1997 (中国語版 博碩文化, 2000) ★本学学生との共著★ |
| 明解C言語入門編 | ソフトバンク, 1998 (中国語版 博碩文化, 2000) |
| 明解C言語入門編 例解演習 | ソフトバンク, 1999 ★本学学生との共著★ |

主な論文

化学プラントの異常時における運転支援システム, 化学工学論文集, Vol.16, No.5, 1990

GUI 環境における上肢障害児のための操作支援インタフェースの開発,

福岡工業大学言語情報工学研究所彙報, Vol.6, 1995

など多数

連絡先等

研究室 1号館7F

はじめに

■ 学生諸君へ贈る言葉 ■

まずは、君たちに以下の言葉を贈ります。

福工大生諸君、

大志を抱き論理能力を磨け。

認識論・弁証法の大家である南郷継正¹⁾の言を紹介します：

現今の若人、とくに高校生に欠けているものが二つある。一つは<大いなる志>であり、一つは<論理能力>である。

いうまでもなく、前者は自己の人生の大いさのかなめとなるものであり、後者は将来己が対象とする事物のもつ構造=性質を、いかばかりにか論理的に究明して一般化し体系化しうるかの能力である。この二つをもちうるべく努めることが、若人にとって歴史性をもって人生に挑める大前提である。

そもそも高校時代は人生の過程中、本来もっとも多感な時期である。それだけに、各人が自由に己が未来を描き、あたかも己が世界を創造しうる能力を一般的にもちうるかのごとき意気軒昂さがあってもよいのだが、大ていはそうならない。それはいったい何ゆえであろうか。端的にいうならば、その教育の<場>に欠けるからである。

本質的にいって、人間はすべてにわたって教育されてはじめて<人間>となりうるのであり、ここに動物との一大区別が存在する。志も論理も直観的成長にまかせてはまともに育つわけはなく、落ちゆく先は小人的君子である。

それゆえ、論理能力の教育は無理としても、せめて大志をまともに育む基盤くらいはほしいものである。それに役立つのが、「個としての大いなる生きざまを描いた文学」であり、こまかい事象に囚われない「壮大なる人類の流れを説いた歴史」である。しかしながら、これすらも見事に与えうる教師を必要としよう。

昔日のわれわれと同様に、今も見事なくその人>が存在しているはずだと思ふのは、これは<見果てぬ夢>なのであろうか。

大志および論理能力という語句について、以下に補足します：

大志 … 人間の目的意識的な存在を事実として認識し、それを目的的に目的意識的にとらえなおした、より見事なる目的意識。

論理能力 … 性質を一般的に法則レベルで把握する能力。

諸君が、福工大を志望した理由はさまざまでしょう。これからの4年間、勉学・クラブ活動・卒業研究などを通じて自己を磨く時間が十分にあります。

大学生ともなると、高校までと比べ、先生たちや御両親の干渉などが激減することになります。したがって、君たち一人一人がどれだけ成長するかは、まさに君たち自身の認識そのものにかかっています。

■ 先輩達の実績 ■

諸君の先輩には、学生時代に専門書を出版している者もいます。

- (1) 柴田望洋 監修・著、赤尾浩¹⁾・松田卓也²⁾ 著
『C言語の鍛練 - 解きながら学ぶC言語 433問 - 』, ソフトバンク, 1996
- (2) 柴田望洋・岡寛文³⁾ 著
『C++への道』, ソフトバンク, 1997
- (3) 柴田望洋・赤尾浩¹⁾・肘井信一⁴⁾・高木宏典⁵⁾・大亀洋⁶⁾ 著
『明解C言語入門編 例解演習』, ソフトバンク, 1999

これらは、現在までに数万冊が発行されており、(2)は海外でも翻訳出版されています。

また、大学院情報工学専攻の学生達は、指導教員の指導の下で、平成6年から平成8年までの3年間だけで、学会発表を計55件（少し古い統計です：平成9年1月31日現在）をこなしているという事実もあります。

君たちの先輩は、この他にも多くの実績をあげています。

■ 成績と就職・大学院進学 ■

大学時代の成績が少しでもいいほうが、就職活動において有利となる可能性は非常に高くなります。また、クラスで上位1/3内であれば、大学院への進学が有利です（推薦を受けることができます）。

一つでも多く“優”が取れるように、大学生活全般を頑張ってください。

-
- 1) 赤尾 浩：福岡工業大学 情報工学科 1996年 卒業（柴田望洋研究室） 吉井システムリサーチ勤務
 - 2) 松田卓也：同大学院 情報工学専攻 1998年 修了（柴田望洋研究室） 日立システムエンジニアリング勤務
 - 3) 岡 寛文：同大学院 情報工学専攻 1999年 修了（柴田望洋研究室） 正興電子勤務
 - 4) 肘井信一：同大学院 情報工学専攻 2000年 修了（柴田望洋研究室）
 - 5) 高木宏典：同 情報工学科 研究生 1999年 修了（柴田望洋研究室） ベースシステム株式会社勤務
 - 6) 大亀 洋：同大学院 情報工学専攻 在学中（柴田望洋研究室）

フロッピーディスクのフォーマット

フロッピーディスクを使用するには、まずフォーマット作業を行う必要があります。これは、田んぼや畑を“耕す”作業だと思ってください。フォーマットしていないフロッピーディスクにデータを読み書きすることはできません。

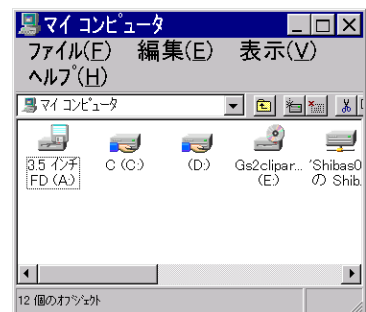
※ ディスクによっては、出荷時にフォーマット済みのものもある。ただし、コンピュータの機種や OS などによって、フォーマット方法が異なります。

- デスクトップ（画面）上の“マイコンピュータ”をダブルクリックします。



※ マイコンピュータのアイコンは、画面左上側にあります。

そうすると、搭載あるいはネットワーク接続されているドライブの一覧表が表示されます。



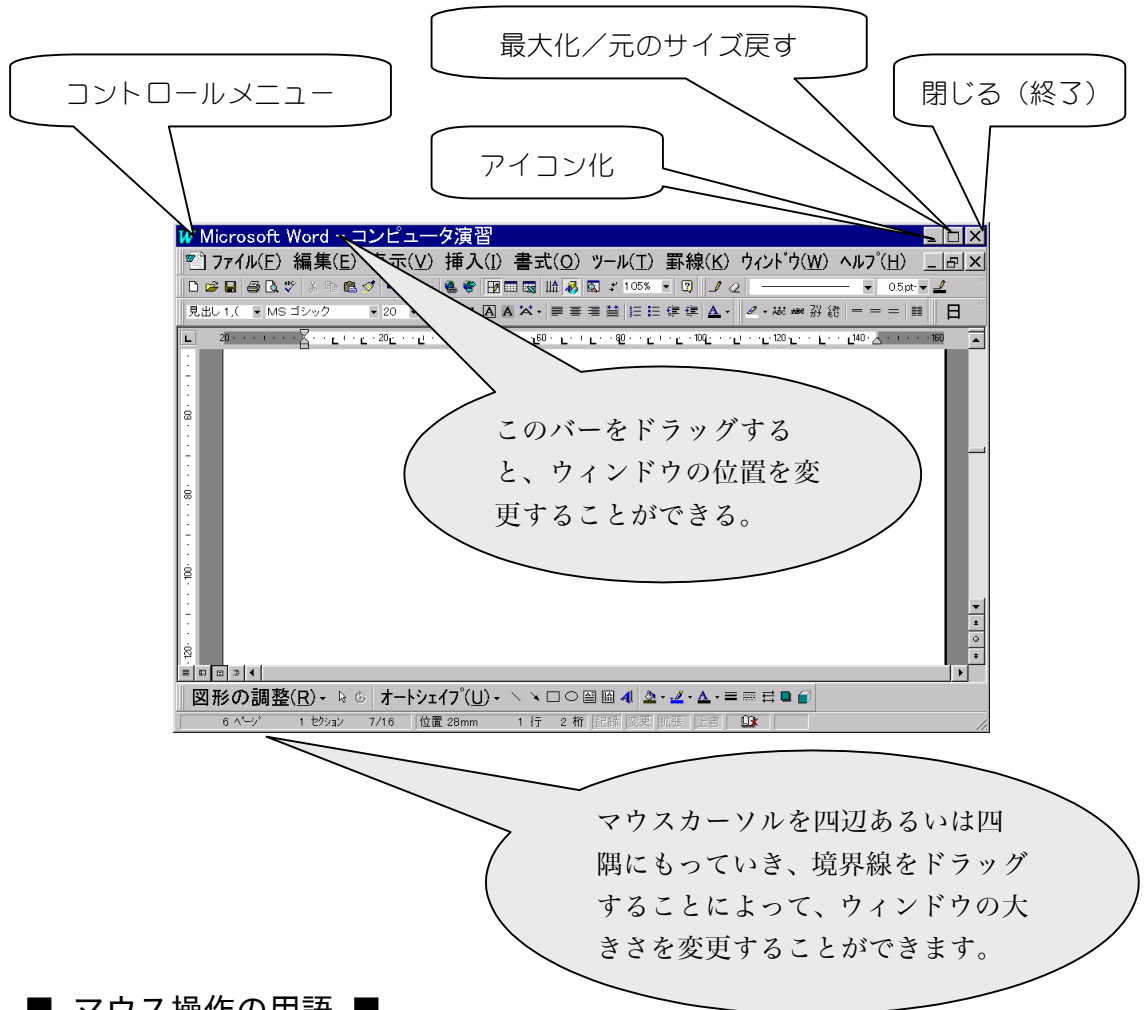
フロッピーディスクのアイコン上で右ボタンをクリックすると、コンテキストメニューが表示されるので、そこから《フォーマット》を選択します。



(以下の作業は実地指導)

コンテキストメニューとは、^{コンテキスト}文脈、すなわち、その時々状況に応じて適切に提示されるメニューのことです。したがって、同じアイコンを右クリックしても、常に同じメニューが表示されるわけではありません。Windows では多用されますので、しっかりと覚えておきましょう。

ウィンドウの操作



■ マウス操作の用語 ■

- クリック ... マウスのボタンを押してすぐ離すこと。
- ダブルクリック ... クリックを連続して2回行うこと。
- ドラッグ ... マウスのボタンを押したままマウスを移動すること。

※ 『電卓』など、大きさを変更できないソフトウェアも存在します。

課題：コントロールメニューをクリックし、何が出来るかを確認してみましょう。

端末室で利用できる二つの OS

■ オペレーティングシステム … OS (*Operating System*) ■

OS は、CPU、主記憶、補助記憶、入出力装置などのリソース (*resource*/資源) を管理し、よりよい環境を利用者に提供するための基本的なソフトウェアです。

■ OS の詳細は、講義『オペレーティングシステム』で学習します。

■ 二つの OS と歴史的背景 ■

(a) MS-Windows98 (*Microsoft Windows 98*) … パソコン用 OS

現在、世界でもっとも有名なソフトウェア会社となっている Microsoft 社が開発 (もともと某会社が作っていたものを買収) した OS である MS-DOS から発展した OS です。

いまや世界中に広がっており、その勢いは誰にも止められそうにありません。今後は、全ての家電製品にも MS-Windows 同様のルック&フィールが採用されるとか (??)。

会長のビル・ゲイツ氏は、高校生の頃から全米的に有名になってしまった、(元) パソコン坊や。世界中のソフト業界が、この坊やの思いつきに左右されているといっても過言ではありません。この坊やの武勇伝などは、いろいろな伝書などで紹介されているので、ソフト業界を目指す人は、読んでおいても損はしません。

■ MS-Windows98 の前身である MS-Windows 3.1 は、MS-DOS という OS 上で動作するソフトウェアであり、OS そのものではありませんでした。また現在、MS-Windows 2000 という OS も存在します (これは Windows 95 や 98 ではなく、Windows NT の後継)。

(b) UNIX … ワークステーション用 OS

AT&T (*American Telephone & Telegraph*) 社のベル研究所の研究者だった Ken Thompson が、ミニコンピュータ DEC PDP-7 を使いこなすために開発し始めたのがきっかけとなって作られた OS で 1970 年頃に完成しました。開発には Dennis Ritchie が加わり、C 言語を産み出すことになりました。

彼らは、MIT の Multics プロジェクトに参加していたため、UNIX は Multics の影響を大きく受けている (多くのことを目指すよりも、1 つのことを目指したため、Multics の [multi=複] を [uni=単] に置き換えた unics に由来するネーミングが行われました)。

1975 年、UNIX は大学などの研究所へ配布が開始され、さらに 1978 年には、C コンパイラを含むようになり、世界中に普及することになりました。1983 年には、Dennis Ritchie と Ken Thompson は、UNIX オペレーティングシステムの設計と開発という功績により、チューリング賞を受賞しています。

■ UNIX と C 言語²⁾ ■

1969 年、大掛かりな Multics プロジェクトは、失敗に終わろうとしていた。目標としていた、高速で使いやすいオンラインシステムはおろか、使えるだけというものすらまったくできあがっていなかったのだ。最後には、何とか動く Multics を作り上げたのは確かだ。しかし IBM が OS/360 で陥ったのと同じ罠に、彼らもやはり引っかかったのだ。彼らが作ろうとした OS はあまりにも巨大で、それに引き換え、使えるハードウェアはあまりにも貧弱だった。Multics は、解決しなくてはならない技術的問題の宝庫だっただけでなく、「小さいことは美しい」という教訓を C に残してくれたのだ。

幻滅と共に Multics プロジェクトから手を引いたベル研の開発者たちは、次に取り組むべき課題を探していた。その一人 Ken Thompson は、別の OS を作成することに熱意を燃やし、何通もの提案書を上司に提出していた (片端からボツになったけど)。承認が出るのを待つ間に、Thompson とその同僚の Dennis Ritchie は、Thompson が作った “Space Travel” というゲームソフトをほったらかしにされていた PDP-7 というコンピュータに移植して楽しんでいた。Space Travel は太陽系を大雑把にシミュレートしており、グラフィック画面上で宇宙船を飛ばし、いろいろな惑星へ着陸することができた。そのために Thompson は猛烈な努力をして、Multics よりはるかに単純で軽快な OS の基礎を PDP-7 に搭載したのだ。OS はすべてアセンブラで書かれていた。1970 年、Brian Kernighan はこの OS を “UNIX” と命名した。もちろんこれは、Multics から学んだ教訓、何をしてはならないかをもじったものだ。

そう、卵と鶏のどちらが先かという点では、間違いなく UNIX が C よりも先に誕生したのである (UNIX の時間情報が、1970 年 1 月 1 日からの通算秒で表されているのもこのためだ — ここからすべてが始まったのである)。アセンブラを使ったことは、困った問題も引き起こした。コードを書くのは骨の折れる仕事だし、デバッグや読むのにも手間がかかる。Thompson は高級言語を使うメリットに惹かれていた。ただし Multics プロジェクトで体験した、PL/I の性能や複雑さはもう御免だった。Thompson はちよつとの間 FORTRAN を試してみたがうまく行かず、B という新しい言語を作成した。これは研究として作られた BCPL という言語を、PDP-7 の 8K ワードのメモリにインタプリタが載るように単純化したものだ。B は決して成功を取めた言語ではない。ハードウェアのメモリ量の制約から、コンパイラではなくインタプリタしか用意できなかった。そのため UNIX 本体を記述するには、B の性能はあまりにも低すぎたのだ。

1970 年になって、新しいマシン PDP-11 への切り替えが行われると共に、型なし言語を開発に使うのには無理があることが明らかになった。性能的な問題もあり、Thompson は PDP-11 用の OS の開発に B 言語を使わず、今回もアセンブラで書くことにした。Dennis Ritchie は PDP-11 の上で、型をサポートし、かつ高性能な “New B” を作ることにした。“New B” — 名前はすぐに “C” に決まった — はインタプリタではなくコンパイラであり、型を導入し、変数を宣言してから使うという仕様になっていた。

歩き煙草は絶対にやめよう

われわれ教員は、《高校訪問》を実施しています。全国の高校を訪問して、進路指導の先生方等に、本学の特長や受験方式・日程などを説明し、

「是非とも福工大受験を生徒達に薦めてください。」

とお願いしているのですね。

ある高校を訪問したときのことで。ちょうど授業間の10分休みの時間でした。ほとんどの生徒は、シャツをズボンに入れずにベロツと出しています。大声で騒ぎまくり、廊下だろうが階段だろうがベランダだろうが、お構いなしにありとあらゆるところに座ったり寝たりしています。

廊下ですれ違ったある先生も、上品という言葉とはかけ離れた存在の人でした。

私は、「こんな生徒達が入学したら困るから、説明せずに帰ろうかなあ。」と本気で感じたものです（進路担当の先生は熱心な方でしたので、きちんと説明してきました … 念のため）。

さて本学はどうでしょう。最も目に付くのが、学生の**歩き煙草**です。学生によっては、吸い殻を廊下にポイと捨てたりしています。

もちろん、本学では、喫煙場所以外での喫煙、歩き煙草は禁止されているのは言うまでもありません。

たとえば、企業の方々が求人のために本学に来訪されたとき、そのような姿を見て、どのように感じられているかを想像してみてください（事実、他大学の先生が、本学のマナーの悪さに驚かれたことがあります）。

自分自身で自らの首をしめてはいけません。

少しでも世間の評判を高め、君たち自身も社会人に向かって成長し、さらには、自分の希望する会社に就職するためにも、歩き煙草を含めて、学内での振る舞いに気を遣って欲しいものです。

UNIX へのログイン

UNIX マシンを起動すると、ログイン画面が現れる。

```
Fukuoka Institute of Technology  
Login   :  
Password:
```

まず、ユーザ名を入力し、**Enter** キーを押します。

```
Fukuoka Institute of Technology  
Login   : s01a1001  
Password:
```

次に、パスワードを入力します。

```
Fukuoka Institute of Technology  
Login   : s01a1001  
Password:
```

※ パスワードの入力時は、キーを打っても、画面には**表示されません**。パスワードは、個々の利用者のみが知っておくべき情報です。パスワードを見て盗まれないように、表示されないのです。

※ Windows からは、TELNET プログラムを通じて、情報処理多センターの UNIX を利用することができます。

パスワードの変更

パスワードは、時折変更しなければなりません。なお、自分にとっては分かりやすく、他人には分かりにくいものとしましょう。abc や fbi、xyz といった誰でも考え付きそうなもの、生年月日や、自分の名前やあだ名 (taku など) などのもつての他です (たとえば、銀行のキャッシュカードの番号を、自分の誕生日にしていると、簡単に悪用されてしまうのと同じことです)。

メモなどもしてはいけません (たとえメモしても、頭で覚えた段階でメモを破棄しましょう。また、このプリントに書きこんだりしないようにしましょう。もしプリントを教室に置き忘れると …)。

```
ipc01% passwd
Changing NIS password for ipc99a1001 on ipc01.
Old password: 古いパスワード ※打ちこんでいる内容は表示されません。
New password: 新しいパスワード ※打ちこんでいる内容は表示されません。
Retype new password: 新しいパスワード ※もう一回打ちこみます。
NIS entry changed on ipc01
ipc01%
```

パスワードを入力後、登録が終了するまで、暫く時間がかかります。慌てず騒がず待ちましょう。

古いパスワードを一回、新しいパスワードを二回打ち込むわけですが、どれかを間違えると、エラーメッセージが表示されます。そのときは、再トライしましょう。

ディレクトリの概念および操作 etc...

ファイルやディレクトリは、端末室やパソコン室で Windows98/NT および UNIX を使う上でも、必須の知識です。ここでは、実用的な操作方法なども交えて解説します。

■ ファイル

データを長期保存することは、OS の重要な役割の 1 つであり、**ファイル (file)** と呼ばれる形式で保存を行います。

<参考>JIS によるファイルの定義

情報処理の目的で、一単位として取り扱われる関連したレコードの集まり

■ ディレクトリ

膨大な数のファイルを 1 元的に管理するのは困難であるため、UNIX や Windows95/NT などの OS では、階層的な構造を持つ**ディレクトリ (directory)** の概念が導入されています。ディレクトリは、複数のファイルをまとめてグループ化するためのものであると考えましょう。ディレクトリは、階層的な構造をもつことができます。

※ 最近の Windows では、ディレクトリでなく“フォルダ”と呼ばれます。

ルートディレクトリ (root directory)

最も上位に位置するディレクトリであり、1 個のみが存在します。

UNIX では / と表し、日本語版 Windows95/NT では ¥ と表します。

サブディレクトリ (sub directory)

階層的なディレクトリ構造において、下位に位置するディレクトリを**サブディレクトリ**あるいは**子ディレクトリ**と呼びます。

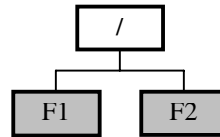
親ディレクトリ (sub directory)

上位に位置するディレクトリ。ルートディレクトリ以外のディレクトリは、必ず 1 個の親ディレクトリを持ちます。

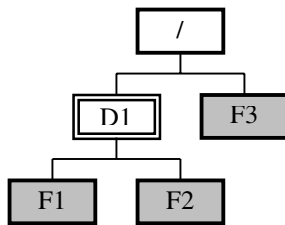
※ ルートディレクトリを含め、各ディレクトリは、任意の個数 (0 個以上) のファイルおよび任意の個数のサブディレクトリを持つことができます。



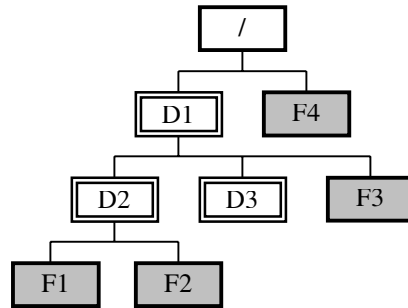
ファイルもサブディレクトリも存在しない。



ルートディレクトリの下にファイル F1 および F2 が存在。サブディレクトリは存在しない。



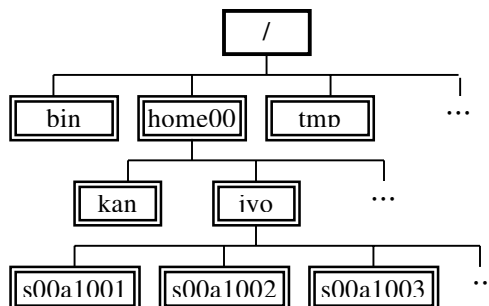
ルートディレクトリの下にディレクトリ D1 およびファイル F3 が存在。サブディレクトリ D1 の下には、ファイル F1 および F2 が存在。
※ディレクトリ D1 の親ディレクトリは、ルートディレクトリ



ディレクトリ D3 の下は空。
※ディレクトリ D2 および D3 の親ディレクトリは、いずれも D1。

■ 本学でのディレクトリ構成

本学センターでは、ルートディレクトリの下に、多くのサブディレクトリが存在する。2000 年 4 月入学の学生用のサブディレクトリが、**home00** であり、その下に、各学科用のサブディレクトリが用意されています。情報工学科用のディレクトリは **jyo** であり、その下に各個人用のディレクトリ **s01a1001**, **s01a1002**, **s01a1003**…が存在します。



■ パス

任意のファイルやディレクトリを表す方式として、以下に示す二つがあります。

■ 絶対パス

ルートディレクトリからの全ての経路で表します。

■ 相対パス

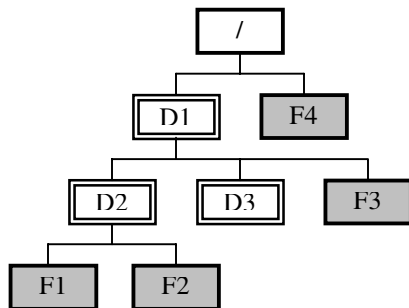
現在作業を行っているカレントディレクトリ (*current directory*) /ワーキングディレクトリ (*working directory*) からの経路で表します。

パスの表現では、以下の記号を利用します。

- .. 親ディレクトリを表す。
- / パス表現の先頭にある場合はルートディレクトリを、
中間にある場合は、ディレクトリ名又はファイル名の区切りを表します。
※日本語版 Windows95/NT では¥を用います。

※ 絶対パスは、ルートディレクトリからの経路で表すため、カレントディレクトリとは無関係に決定します。

※ 相対パスは、カレントディレクトリからの経路で表すため、カレントディレクトリに依存します。



左図において、各ファイルの絶対パス、相対パスを考えてみましょう。

絶対パス

ファイル F1 /D1/D2/F1
 ファイル F3 /D1/F3
 ファイル F4 /F4
 ディレクトリ D3 /D1/D3

相対パス

| | |
|-------------------------|----------------------|
| カレントディレクトリが D2 であるとき | カレントディレクトリが D1 であるとき |
| ファイル F1 F1 | ファイル F1 D2/F1 |
| ファイル F3 ../F3 | ファイル F3 F3 |
| ファイル F4 ../../F4 | ファイル F4 ../F4 |
| ディレクトリ D3 ../D3 | ディレクトリ D3 D3 |

第二種情報処理技術者試験【平成7年度秋】午前・問49

MS-DOSやUNIXにおけるディレクトリ構造に関する記述のうち、正しいものはどれか。

- ア サブディレクトリの下に更にサブディレクトリを置くことができる。
- イ ファイルの下にサブディレクトリを置くことができる。
- ウ ファイルをルートディレクトリの下に置くことはできない。
- エ ルートディレクトリの下に更にルートディレクトリを置くことができる。
- オ ルートディレクトリの下にはサブディレクトリを一つ以上置かなければならない。

第二種情報処理技術者試験【平成8年度秋】午前・問31

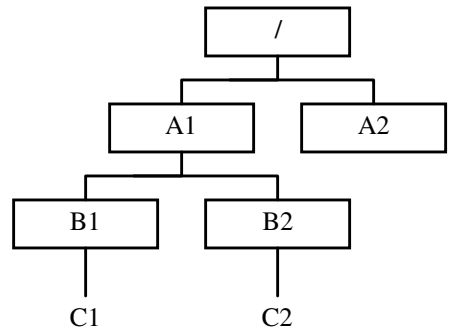
階層構造のディレクトリを用いたファイル管理を行うオペレーティングシステムにおいて、ファイルが置かれているディレクトリを指すときに指定するものはどれか。

- ア 拡張子 イ サブディレクトリ ウ パス
- エ ルートディレクトリ オ ワイルドカード

第二種情報処理技術者試験【平成9年度春】午前・問46

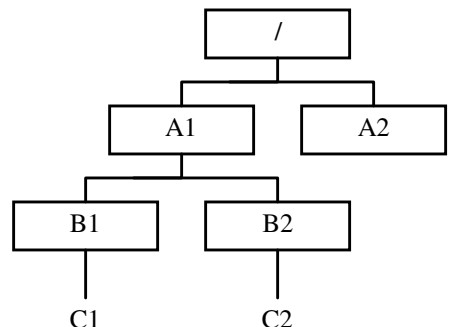
次の階層ファイルシステムにおいて、カレントディレクトリがB1であるとき、ファイルC2を指す相対パスはどれか。ここで、パスの表現において“..”は親ディレクトリを表し、“/”はパス表現の先頭にある場合はルートディレクトリを、中間にある場合は、ディレクトリ名又はファイル名の区切りを表す。また、図中の□はディレクトリを表すものとする。

- ア ../A1/B2/C2 イ ../B2/C2
- ウ A1/B2/C2 エ /A1/B2/C2

**第二種情報処理技術者試験【平成12年度春】午前・問47**

図の階層ファイルシステムにおいて、カレントディレクトリがB1であるとき、ファイルC2を指す相対パスはどれか。ここで、パスの表現において“..”は親ディレクトリを表し、“/”はパス表現の先頭にある場合はルートディレクトリを、中間にある場合は、ディレクトリ名又はファイル名の区切りを表す。また、図中の□はディレクトリを表すものとする。

- ア ../A1/B2/C2 イ ../B2/C2
- ウ A1/B2/C2 エ /A1/B2/C2

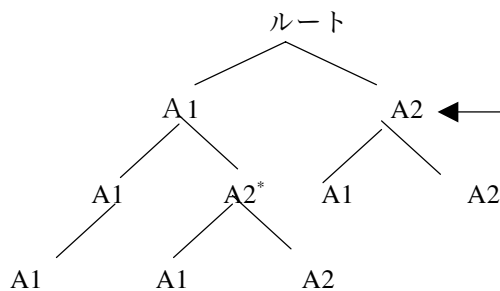


第二種情報処理技術者試験【平成 11 年度春】午前・問 50

複数個のディレクトリ A1, A2 が図の構造で管理されている。各ディレクトリには、ファイル f が存在する。*印のディレクトリ (カレントディレクトリ) から矢印のディレクトリ配下のファイル f を指定する方法はどれか。

ここで、ファイルの指定方法は、次によるものとする。

- ① ファイルは“ディレクトリ名¥ … ディレクトリ名¥ファイル名”のように、経路上のディレクトリを順に“¥”で区切って指定する。
- ② カレントディレクトリは“.”で表す。
- ③ 1 階層上のディレクトリを“..”で表す。



ア .¥A2¥f

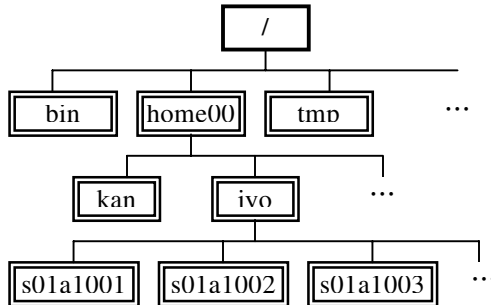
イ ..¥.¥A¥f

ウ ..¥A1¥..A2¥f

エ ..¥A2¥f

■ ホームディレクトリ

マルチユーザの利用を想定した UNIX では、利用者各個人専用のディレクトリが用意されて、そのディレクトリをホームディレクトリ (*home directory*) と呼びます。



2001 年入学の情報工学科の学生のホームディレクトリを絶対パスで表すと、次のようになります。

```
/home00/jyo/s01a1001
```

```
/home00/jyo/s01a1002
```

```
/home00/jyo/s01a1003
```

:

なお、ログインした際には、ワーキングディレクトリ (カレントディレクトリ) は、ホームディレクトリとなります。

確認

ログインしたら、すぐにワーキングディレクトリを表示してみましょう。

```
ipc01%> pwd  
/home00/jyo/s00a1001
```

ホームディレクトリを表す記号は、~です (「チルダ」と呼びます)。

UNIX では、パラメータを与えずに cd コマンドを実行すると、ワーキングディレクトリは、ホームディレクトリへと移ります。

演習

C 言語用のディレクトリを作成してみよう。

```
ipc01%> pwd
/home01/jyo/s01a1001
ipc01%> mkdir C
ipc01%> ls
C
```

※ c は大文字で!!

※ ファイルの一覧が表示される

演習

(続けて) ホームページ用のディレクトリを作成してみよう。

```
ipc01%> mkdir WWW
ipc01%> ls
C WWW
```

※ www は大文字で!!

※ ファイルの一覧が表示される

演習

(続けて) 作成したディレクトリの詳細を見てみよう。

```
ipc01%> ls -l
drwxr-xr-x 2 s01a1001 512 Jun 20 10:51 C/
drwxr-xr-x 2 s01a1001 512 Jun 20 10:52 WWW/
```

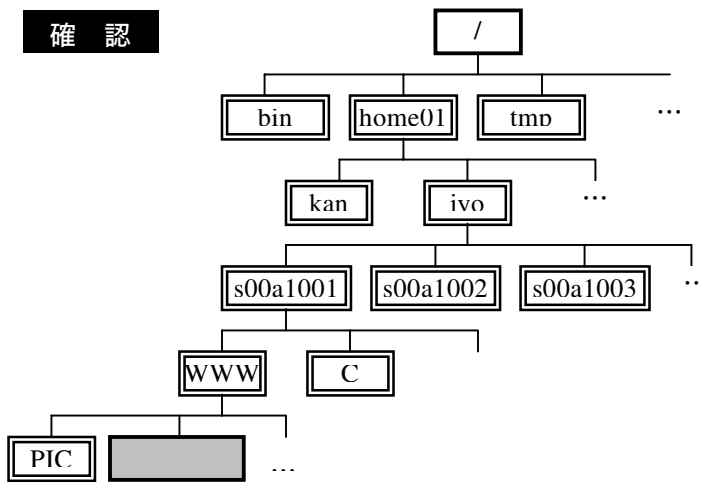
演習

(続けて) WWW ディレクトリの下に画像ファイル保存用の PIC ディレクトリを作成する。

```
ipc01%> mkdir WWW/PIC
ipc01%>
```

※ WWW/PIC は大文字で!!

※ディレクトリの作成を一度行くと、自分で消さない限り残るので、同じ命令を2度やったら、エラーが出ることに注意しよう。

確認

現在、諸君のホームディレクトリの下には、少なくとも、以下の二つのサブディレクトリが作られているはずである。

| | |
|---------------------|---------------|
| サブディレクトリ www | ホームページ用ディレクトリ |
| サブディレクトリ c | C 言語用ディレクトリ |

確認

ホームディレクトリの下での WWW ディレクトリへ移動する方法を確認する。

絶対パスを利用

```

ipc01%> cd /home01/jyo/s01a1001/WWW
ipc01%> pwd
/home01/jyo/s01a1001/WWW

```

ホームディレクトリからの経路を利用

```

ipc01%> cd ~/WWW
ipc01%> pwd
/home01/jyo/s01a1001/WWW

```

相対パスを利用

```

ipc01%> cd WWW
ipc01%> pwd
/home01/jyo/s01a1001/WWW

```

※ この方法は、ワーキングディレクトリがホームディレクトリ以外であれば、無効である。したがって、ホームディレクトリからの経路を利用したほうが確実である。

確認

ホームディレクトリへ移動後、その下の WWW ディレクトリへ移動。その後、ホームディレクトリの下の子ディレクトリへ移動する。

例 1

```
ipc01%> cd
ipc01%> pwd
/home01/jyo/s01a1001/www
ipc01%> cd www
ipc01%> pwd
/home01/jyo/s01a1001/www
ipc01%> cd ../C
ipc01%> pwd
/home01/jyo/s01a1001/C
```

例 2

```
ipc01%> cd
ipc01%> pwd
/home01/jyo/s01a1001/www
ipc01%> cd ~/www
ipc01%> pwd
/home01/jyo/s01a1001/www
ipc01%> cd ~/C
ipc01%> pwd
/home01/jyo/s01a1001/C
```

×例 3×

次の例は間違いである (WWW ディレクトリの下には C ディレクトリが存在しないため)。

```
ipc01%> cd
ipc01%> pwd
/home01/jyo/s01a1001/www
ipc01%> cd www
ipc01%> pwd
/home01/jyo/s01a1001/www
ipc01%> cd C
C: No such file or directory
```

確認

Cディレクトリの下に list0101.c というファイルを作成しているとする。その内容を表示する。

例1 ホームディレクトリからの経路で表す。

```
ipc01%> cat ~/C/list0101.c
```

例2 絶対パスで表す。

```
ipc01%> cat /home99/jyo/s01a1001/C/list0101.c
```

例3 相対パスで表す (ワーキングディレクトリがホームディレクトリであるとする)。

```
ipc01%> cat C/list0101.c
```

この方法は、ワーキングディレクトリがホームディレクトリ以外であれば無効。

例4 相対パスで表す (ワーキングディレクトリがCディレクトリであるとする)。

```
ipc01%> cat list0101.c
```

この方法は、ワーキングディレクトリがCディレクトリ以外であれば無効。

“～じゃないですかあ” 言葉はやめよう

最近テレビを見ていると、芸能人ばかりか、比較的教養のある人までもが、会話の最後に

“～じゃないですかあ。”

を連発している。これは、非常に聞きづらいばかりか、目下の人に言われると、馴れ馴れしく感じて、場合によっては腹立たしくも感じる言葉である。

そもそも、

「あのとき私はきちんと確認したじゃないですか。」

という例からもわかるように、この表現は、一応は丁寧ではあるものの、かなりキツイ感じも与える。あるとき、私の研究室を訪ねてきた学生が、

- … そもそも僕はアルバイトをしてるじゃないですか。
- … C++って、Cに比べて難しいじゃないですか。
- … 他の本と違って、先生の本は、分かりやすいじゃないですか。

おそらく僅か数分間に、百連発?!? (ちなみに、初めての学生であって、アルバイトしている話しも初耳)。

目上の人にこのような言葉づかいをすると、

- 馴れ馴れしく、失礼な感じを与える。
- ただ意見を述べているだけのつもりでも、同意を強要しているように取られる。

など、悪いところだらけである。このような言葉遣いは慎もう。芸能人の真似をするな!! 普段から練習しておかないと、**就職活動の面接のときにポロッと出てしまうぞ。**

- … そもそも大学って高校までと違って自由じゃないですかあ。
- … 最近はインターネットがはやっているじゃないですかあ。
- … 一次面接では、この問題を解くように言われたじゃないですかあ。

そうすれば必ずやマイナスの印象を与えるだろう。人間イザというときは、“普段”が出てしまうものである。普段から、気を付けておこう。

文献の引用ということについて

論文、レポートなど、ドキュメントを作成したときは、いかなる場合でも、参考とした文献や、引用した文献を必ず記さなければならない。ここでは、その重要性について、二つの文献から引用する。

S

科学論文においては適切な reference (文献) を cite (引用) するのは非常に大切な事であって、これは我田引水にならぬ様常にフェアな態度で行う様心がけるべきである。Reference を使うということは情報やアイデアの系譜をはっきりさせる事であり、自分の主張を正当化する根拠を示す事である。欧米ではオリジナルなアイデアを誰が最初に出したのかを明確にする事に非常に神経を使っていて、うっかり引用すべき文献を落としたとすると、思わぬ敵を作ることになる。日本ではこの様な訓練はしない様だが、米国では高校位から、自分のオリジナルなアイデアと他からの受売りのアイデアの区別をはっきりさせる訓練をしていて、普段書くレポート等でも常に reference の citation をさせる様にしている。読者は plagiarism という単語を御存知だろうか。これは他人のアイデアを自分のもののように見せかけて提出することで、非常に悪い事だとされている。Plagiarism には代理人に書いて貰ったレポートを提出する事から、他人のアイデアを reference citation せずに自分の論文に書き込むことまで含まれていて、大学によっては plagiarism で捕まると即時退学という所も多いから恐ろしい。英語世界は reference に関してはこれ程神経質なのだから、論文を書く時には相当注意する必要がある。またもっとも現実的な面では、引用しなかった文献の著者が referee になって物言いが付き、出版が遅れるなどという事態も起こる。

日本物理学会編『科学英語論文のすべて』, pp.56-57, 丸善, 1984

※網掛けは引用者による。

■ 参考 ■ 人気映画「刑事コロンボ」では、他人のレポートを盗作した学生が、教授に“退学にしてやる”と言われ、逆恨みして教授を殺すという事件が扱われていた。レポートを写せば即刻退学なのだ!!

S

ここで、本物の初心者にたいして一言を念のために申し添えておく。ここで借りものの一章・一節・一文とは、けっしてその書名・著者名を堂々と明らかにしての他の先達からの引用のことではなく、他人の著書から無断で借りてきているにもかかわらず、こっそりとあたかも自分自身の文であるかのごとくに用いることや、自分の究明した論理や理論でもないものを、さも自己の研究成果であるかのごとくに知らぬ顔をして自分の文章として改竄して用いる輩の文章のことである。したがって、読者諸君がなんらかの機会に自らの文章を公けにされるばあいには、他人の文章は出典を明らかにすることを原則とし、出典が不明のばあいは著者不明と認めることが肝心である。

(次頁に続く)

他人の文章をこっそりと借用したばあいには、いかに巧みに自らの論として用いたつもりであっても、かならずバレるものなのである。諺に曰く、<隠すより現るるはなし>であって、他人の論理はまずは己れの論理に見事には適合しないものと覚悟すべきである。それよりもなによりも、他人の血と汗と涙の結晶である理論や論理をあたかも己れ自身の発見に関わるかのごとくに用いることは、これは盗人にも似て卑劣そのものである。

もちろん、他人の発見、研鑽に関わる理論や論理は、たしかにその人のものではあるが、その人が一度筆をとってその発見・研鑽に関わる理論や論理を公表すれば、その理論なり論理なりはその人のものでありながら社会的となり、人類の文化遺産として万人に供されることになるのである。

これはたとえば、ピタゴラスの定理やオームの法則のごとくにである。しかしながら、このオームの法則やピタゴラスの定理のごとくに発見者の手からまったく離れて独立的に一人歩きを始めた理論や法則などのばあいは改めて出典を明示する必要はない。なぜならこれは天下周知の事実であるだけに、ことさらのなにのなにがしが手になるものと明言せずとも誰にでもわかっているからであり、知らないほうが無知として恥になるくらいの公的法則であるからである。

だが、そのような周知のものとは違っていまだ社会的に一般性をもつレベルにいたらない理論や論理のばあいには、たとえどんなに面倒であっても、それを用いるばあいには、引用という形式であれ、自分の文章の一部として用いるばあいであれ、自分の論の展開として用いるばあいであれのいずれを問わず、筆者は堂々とその発見者・研鑽者を記し、出典を明示する形で敬意を表するのを大原則とするものである。それがどうしても嫌ならば、事は簡単である。その理論なり論理なりをこっそりと用いて盗作呼ばわりをされるか、絶対に自らのものとして用いることをしないかである。再言しておくが、堂々と用いるとは、あくまでも出典を明示して引用すると明言することであり、けっしてそれ以外ではない。

私事に関わるが、私も公けに筆をとって二十年、当然のこととして先達の価値ある論文を引用するを常態としてきたが、逆にまたいろいろな人たちの文章の中に盗用という形の私の論を見出してきていることである。だが、私はほとんどその盗用を論じたことは公けにはない。

なぜかというに、あまりに私からの盗用が明白すぎたからである。具体性でいえば、その論文!? は私からの盗用の箇所・論理だけが輝いて見え、その盗んだ御当人の論の展開箇所となると、これはもう吹きだすことなしにはとうてい読めたシロモノではなかったからである。それにしても、どうして一言断れないものであろうか。それら盗用者の認識をあれこれ想像するうちに、もの悲しくなってきたことである。

南郷継正『武道の科学』, pp.22-24, 三一書房, 1991

参考文献

■ 本文中で引用した文献 ■

- 1) 南郷継正 『武道への道 武道をとおしての教育論』, 三一書房, 1979
- 2) Peter van der Linden, Expert C Programming - DEEP SECRETS, 1994
梅原系訳 『エキスパート C プログラミングー知られざる C の深層』, アスキー, 1996
- 3) 日本物理学会編 『科学英語論文のすべて』, 1984
- 4) 南郷継正 『武道の科学』, 三一書房, 1991

■ その他の参考文献 ■

- 5) 山口和紀 『The UNIX Super Text 【上】』, 技術評論社, 1990
- 6) 南郷継正 『武道講義 第一巻 武道と認識の理論 I』, 三一書房, 1990