

UNIX初級講習会

九州大学情報基盤センター

本資料は以下のURLで閲覧できます。
<http://www.cc.kyushu-u.ac.jp/scp/lecture/unix1/>

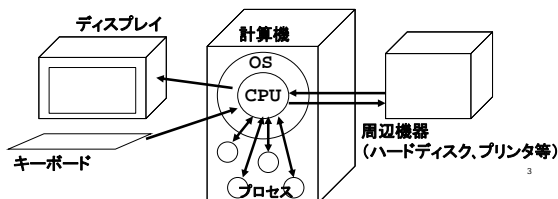
スケジュール

- UNIXの概要
- 計算機への接続, ログイン, ログアウト
- ファイル操作
- テキストファイルの編集
- プログラムの翻訳, 実行
- その他のコマンド
- ウィンドウシステム(X Window)

2

UNIX とは

- OS (Operating System) の一種
 - OS: 計算機の管理を行う基本ソフトウェア
 - キーボードからの入力, ディスプレイへの出力, ファイルの読み書き, プロセスの管理など
 - 代表的な OS:
UNIX, Windows, DOS, MacOS 等



UNIX の特徴

- マルチユーザー
 - 同時に複数のユーザーが同じ計算機を利用可能
 - 比較的厳格なユーザー管理 (ユーザー ID, パスワード)
- マルチタスク
 - (見かけ上) 同時に複数の処理を実行可能

4

情報基盤センターの 研究者用計算機システム

- kyu-vpp.cc.kyushu-u.ac.jp
 - スーパーコンピュータ Fujitsu VPP5000/64
- kyu-cc.cc.kyushu-u.ac.jp
 - 汎用UNIXサーバ IBM eServer p5 model 595 / 570
- wisdom.cc.kyushu-u.ac.jp
 - 多目的サーバ Sun Microsystems Sun Blade2500
- kyu-ss.cc.kyushu-u.ac.jp
 - スカラー並列サーバ Compaq GS320
- vhsun.cc.kyushu-u.ac.jp
 - アプリケーションサーバ Fujitsu GP400S

5

情報基盤センターへの利用登録

- アカウント
 - 情報基盤センターの研究者用計算機利用権
 - 以下の Web ページ, もしくは共同利用掛(5階)で「利用申請書」を入手し記入後提出
<http://www.cc.kyushu-u.ac.jp/scp/users/download.html>
 - 記入方法は以下の Web ページを参照
<http://www.cc.kyushu-u.ac.jp/scp/users/riyou1.html>
 - 1~2週間程度で「利用承認書」送付
 - 利用承認書: アカウントに関する情報
 - ユーザー ID
 - 初期パスワード

6

情報基盤センターの 問い合わせ窓口

- ホームページ: <http://www.cc.kyushu-u.ac.jp>
- 電子メール: request@cc.kyushu-u.ac.jp
- 利用の手引き: 共同利用掛(5階)
- マニュアル: 図書室

7

資料

- UNIX利用法
<http://www.cc.kyushu-u.ac.jp/scp/users/pdf/unixguide.pdf>
 - UNIXの基本コマンド等
- スーパーコンピュータ VPP5000 利用法
<http://www.cc.kyushu-u.ac.jp/scp/system/general/VPP5000/>
 - プログラム開発やソフトウェア利用に関する情報
- 高性能演算サーバ並列プログラミング入門
<http://www.cc.kyushu-u.ac.jp/scp/users/pdf/parallel.pdf>
 - OpenMP, MPI及び自動並列化による並列処理の基礎

8

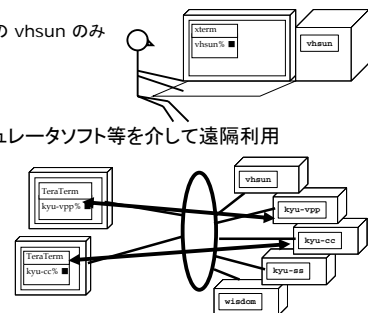
スケジュール

- UNIXの概要
- 計算機への接続, ログイン, ログアウト
- ファイル操作
- テキストファイルの編集
- プログラムの翻訳, 実行
- その他のコマンド
- ウィンドウシステム(X Window)

9

情報基盤センターの UNIX 利用

- 直接
 - センター 2階の vhsun のみ
- PCの端末エミュレータソフト等を介して遠隔利用
 - 通常はこちら。



10

PCからの接続

- 「ssh対応端末エミュレータ」ソフトを利用
 - TeraTerm+TTSSH:
 - ネットワークから無料で入手可能
 - 設定方法は下記の URL で紹介
- <http://www.cc.kyushu-u.ac.jp/scp/system/general/GP7000F/telnet.win/>
- 「Xサーバ」ソフトを利用
 - Astec-X等

11

セッション: UNIX の利用開始から終了まで

- UNIXへの接続
- UNIXへのログイン
 - パスワードによる認証後、セッション開設
- 作業
 - ファイル操作、プログラムの作成、翻訳、実行
- UNIXからのログアウト
 - セッション終了

12

初回アクセス時の注意

- 必ずログインしてパスワードを変更する
 - passwdコマンド
 - kyu-cc では初回ログイン時にパスワードの変更を求められる。

```
Your password has expired; Choose a new one.  
You must change your password now.  
Changing password for "s70080a"  
s70080a's Old password:
```

 - 初期パスワード入力後、新パスワードを2回入力
 - 変更後、一度接続が切れるので利用を続ける場合は再度ログインする。
 - パスワードを変更するまでファイル転送 (WinSCP, sftp等)は不可

13

パスワードの重要性

- ユーザーの認証に利用
 - 他人に知られた場合の危険
 - ファイルの破壊
 - そのユーザーになりすましてメールを送付
 - etc.
- パスワードの管理
 - 他人に知られないようにする
 - 初期パスワードの変更: passwd コマンド
 - メモ等、他人が読める形で残さない
 - 類推が容易なパスワードは避ける

14

UNIX のコマンド入力

- プロンプト = コマンド入力待ち
`kyu-cc% █`
- 基本的なコマンドの形式

コマンド名	オプション	引数
pwd		
cd		test
ls	-l	test
- 打ち間違えたら
 - BackSpace(後退), C-u (クリア) で訂正
- 改行キーを押して実行

15

サーバからのログアウト

- exit コマンドによりセッション終了
- Windowsのログオフやシャットダウンをする前に必ずUNIX からログアウト
 - 実行中のプログラムがそのまま残る可能性あり。

16

Tera Term で接続 - 教育用システムの場合

- Windows のデスクトップで [TeraTerm Pro]アイコンをダブルクリック
- Tera Term New Connection ウィンドウで
 - Hostに kyu-cc.cc.kyushu-u.ac.jp を入力
 - Service の SSH をチェックして [OK]
- SECURITY WARNINGウィンドウがでたら
 - Add this machine ... をチェックして [Continue]
 - エラーが出て、慌てず [OK]

17

ログイン

- SSH Authentication ウィンドウで
 - ユーザID, パスワードを入力
 - Use plain password to log in をチェックして[OK]
- 初期パスワード変更後、再度TeraTermを起動してログイン

18

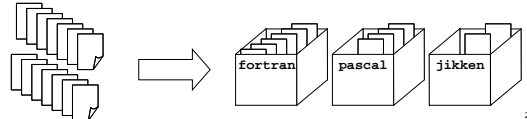
スケジュール

- UNIXの概要
- 計算機への接続, ログイン, ログアウト
- **ファイル操作**
- テキストファイルの編集
- プログラムの翻訳, 実行
- その他のコマンド
- ウィンドウシステム(X Window)

19

UNIXにおけるファイルの整理

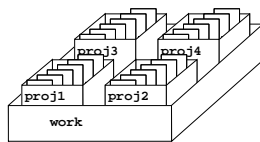
- ファイルの数が多くなると
 - 目的のファイルを探すのが面倒
 - 新しく作るファイルの名前を考えるのが面倒
- ディレクトリ(= ファイル分類用の箱)を利用して整理
 - ファイルを目的別に分類して格納することにより, 管理(探索, 再利用, 削除等)を容易にする.
 - Windows のフォルダと同じ働き.



20

ファイルの分類法

- 自分が管理しやすいように.
 - 例えば
 1. 研究に関連するファイルは work ディレクトリの中に置くことにする.
 2. さらに, workディレクトリの中にプロジェクト毎のディレクトリを用意し分類する.

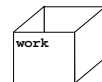


21

ディレクトリの作成 mkdir コマンド

- 使い方:
`mkdir` 作成したいディレクトリ名
- 例: workディレクトリの作成
`% mkdir work`

空白(スペース)を
忘れない

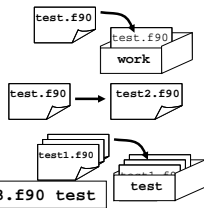


22

ファイルの移動 mv コマンド

- 使い方:
`mv` 移動元 移動先

- 例1: test.f90 を workディレクトリに移動
`% mv test.f90 work`
- 例2: test.f90 の名前を test2.f90 に変更
`% mv test.f90 test2.f90`
- 例3: test1.f90, test2.f90, test3.f90 を test ディレクトリに移動
`% mv test1.f90 test2.f90 test3.f90 test`



23

ファイルとディレクトリの構造

- 準備:
ディレクトリにファイルが入っている状態をこのように表すことにする.

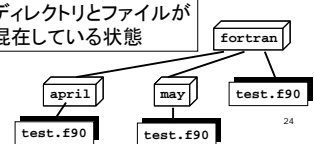
1つのディレクトリに
1つのファイルが
入っている状態



複数のファイルが
入っている状態



ディレクトリとファイルが
混在している状態

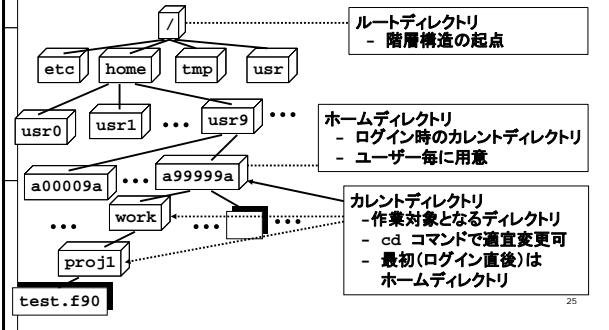


24

サーバ全体のファイル構造

□ 逆さにした木のような構造

- 一番上にルート(根)ディレクトリ



25

ホームディレクトリとカレントディレクトリ

- カレントディレクトリ(=現在"いる"ディレクトリ)を基準としてファイルやディレクトリの場所を指定する。
- ログイン直後はホームディレクトリ(=自分専用のディレクトリ)に"いる"。

- 例1: カレントディレクトリにディレクトリ作成
`% mkdir work`
- 例2: カレントディレクトリの test.f90 をカレントディレクトリの下 work ディレクトリに移動
`% mv test.f90 work`

26

別のディレクトリに移る cd コマンド

□ 使い方:

cd [移動先ディレクトリ]

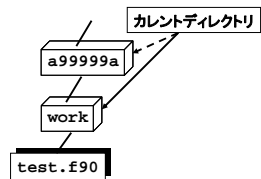
- 別のディレクトリに移る = カレントディレクトリの変更
- 移動先ディレクトリを指定しない場合、ホームディレクトリに移る。

- 例1: workディレクトリに移る。

`% cd work`

- 例2: ホームディレクトリに移る。

`% cd`



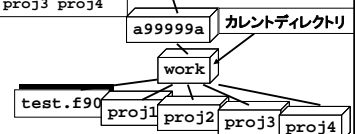
27

別のディレクトリに移って作業

- 作業対象のファイルやディレクトリがあるディレクトリに移って作業する。

- 例: work ディレクトリに移って、各プロジェクトのディレクトリを作成する。

`% cd work`
`% mkdir proj1 proj2 proj3 proj4`



ディレクトリを移らない場合、キー入力が大変:

`% mkdir work/proj1 work/proj2 work/proj3 work/proj4`

28

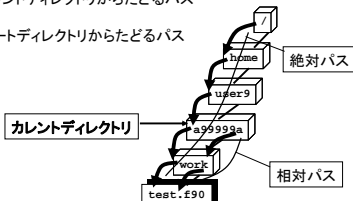
パス: ファイルやディレクトリの場所

- 目的のファイルやディレクトリまでたどる際の経路(path)にあるディレクトリを列挙

- 二通りの記述方法(どちらでも良い)

□ 相対パス: カレントディレクトリからたどるパス

□ 絶対パス: ルートディレクトリからたどるパス



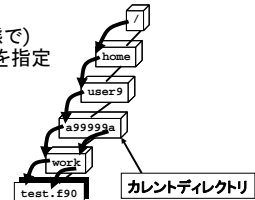
29

パスの記述方法: 「ディレクトリの下」を / で記述

- 例: (今ホームディレクトリにいる状態で) workディレクトリの下 test.f90 を指定

- 相対パス (カレントディレクトリの下) workの下 test.f90 work/test.f90

- 絶対パス ルートディレクトリの下 home の下の usr9 の下の a99999a の下の work の下の test.f90 /home/usr9/a99999a/work/test.f90
- ルートディレクトリも / で記述する。



30

パスの記述に用いる特殊な記号

- 一つ上のディレクトリ .. (ピリオド二つ)

- 例: カレントディレクトリの一つ上に移る

```
% cd ..
```



- カレントディレクトリ . (ピリオド一つ)

- 例: 一つ上のディレクトリの test.f90 をカレントディレクトリに移動する.

```
% mv ../test.f90 .
```

- ホームディレクトリ ~

- 例: ホームディレクトリの work ディレクトリに移る.

```
% cd ~/work
```

31

ファイル名の補完 *

- ファイル名の一部だけを入力して操作できる

- 例: ファイル名の最初が test で、末尾が f90 であるファイル全てを test ディレクトリに移

```
% mv test*f90 test
```

- 便利で危険な機能

- 思いもよらないファイルまで操作してしまうことがあるので注意.

32

カレントディレクトリの確認 pwd コマンド

- 使い方:
pwd

- 使用のタイミング:

- cd コマンドで移動した後の確認.
- 重要な操作(ファイルの削除や移動等)を行う前の確認.

33

ファイルの一覧表示 ls コマンド

- 使い方:
ls [オプション] [ファイル名またはディレクトリ名]

- 例1: カレントディレクトリの全ファイル、ディレクトリを表示

```
% ls
```

- 例2: work ディレクトリの中の全ファイル、ディレクトリを表示

```
% ls work
```

- 例3: カレントディレクトリの中の全ファイル、ディレクトリの詳細情報を表示

```
% ls -l
```

- 例4: 指定したファイルの詳細情報を表示

```
% ls -l ファイル
```

34

ファイルの複製 cp コマンド

- 使い方:
cp [オプション] 複製元 複製先

- 例1: test.f90 を test2.f90 という名前で複製

```
% cp test.f90 test2.f90
```

- 例2: test.f90 と test2.f90 を work ディレクトリにコピー

```
% cp test.f90 test2.f90 work
```

- 例3: (backup ディレクトリがまだ存在しなければ) work ディレクトリ全体を backup という名前で複製

```
% cp -r work backup
```

- 例4: (backup ディレクトリが既に存在すれば) work ディレクトリ全体を backup ディレクトリの下に複製

```
% cp -r work backup
```

35

ファイルの削除 rm コマンド

- 使い方:
rm [オプション] 削除するファイルやディレクトリ

- 例1: test.f90 と hello.f90 を削除

```
% rm test.f90 hello.f90
```

- 例2: work ディレクトリとその下の全てのファイルを削除

```
% rm -r work
```

- 例3: 削除の前に確認

```
% rm -ir work
```

36

UNIXのファイルに関する注意

- 削除したら(原則として)復元不可能
 - 「ごみ箱」は存在しない.
- コピーや移動で既に存在するファイルが
コピー先や移動先になっている場合上書きされる.
 - この場合も復元不可能
- 各ファイルについて削除や上書きの前に確認する `-i` オプションを利用する.


```
rm -i -r ディレクトリ
mv -i ファイル1 ファイル2 ファイル3 ディレクトリ
cp -i ファイル1 ファイル2 ファイル3 ディレクトリ
```

37

ファイルの内容表示 less コマンド

- 使用方法:
 - less ファイル名
 - ファイルの内容を1ページずつ表示
 - 閲覧中は以下のキーを利用
 - 次のページへ: SPACE もしくは f
 - 前のページへ: b
 - 1行下へ: Enter もしくは j
 - 1行上へ: k
 - 閲覧終了: q

38

Tcsh: 高機能シェル

- 通常のシェルに様々な機能を追加し、より便利になったシェル.
 - 補完機能
コマンドやファイル名を途中で入力し、TABキーを押すと残りを補完する.
 - コマンド履歴の呼び出し機能
今まで入力したコマンドを矢印キーの上下で呼び出す.
 - コマンド行編集機能
入力中のコマンドの書き換え、カットアンドペースト等.
- 起動と終了
 - 起動: tcsh
 - 終了: exit
- ログインシェルの変更
 - chsh コマンド
 - 次のログインから有効.

```
kyu-cc% chsh
current available shells:
/usr/bin/sh
/usr/bin/ksh
/usr/bin/csh
/usr/bin/bash
/usr/bin/tcsh
current login shell:
/usr/bin/csh
change (y/n)? >y
to ? >/usr/bin/tcsh
rm: all.time を削除しますか?y
```

39

Tcshの利用法

- 通常のコマンドに加え、以下の機能を利用可能

キー操作	機能
→	カーソルを一つ右に移動
←	カーソルを一つ左に移動
C-a	左端に移動
C-e	右端に移動
C-u	全部削除
C-k	カーソルから右側を切り取り
C-y	直前に切り取った部分を貼り付け
↑	実行したコマンド履歴の古い方へ
↓	実行したコマンド履歴の新しい方へ
TABキー	コマンドやファイルの補完

40

実習1

実習1: ログインとファイル操作

- ログイン
- ファイル操作
- ログアウト

41

スケジュール

- UNIXの概要
- 計算機への接続, ログイン, ログアウト
- ファイル操作
- テキストファイルの編集
- プログラムの翻訳, 実行
- その他のコマンド
- ウィンドウシステム(X Window)

42

Emacs(イーマックス)

- UNIXにおける標準的なエディタ
- プログラムやジョブスクリプトなどの入力に利用。
 - 実は、メールの読み書きやWebページの閲覧も可能.

43

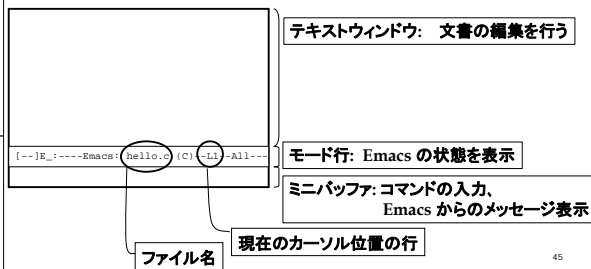
Emacs

- 使い方
emacs ファイル名
 - 例: hello.c の編集
`% emacs hello.c`
- 指定したファイルがまだ存在しない場合は新規作成.
- 既に存在する場合は読み込んで修正。
 - 修正後保存すると、修正前のファイルが
ファイル名~
という名前で残る.

44

Emacs の画面

- 起動後



45

Emacs のバッファ

- Emacs が扱うファイルのイメージ
 - 実体ではない
 - 編集後、保存動作によってファイルに反映

46

Emacsの操作

- Ctrlキーを押しながら... の操作
 - C-x ... Ctrl キーを押しながら x を押す.
 - C-BackSpace ... Ctrl キーを押しながら BackSpaceキーを押す

47

Emacsのコマンド

キー操作	意味
C-f または →	カーソルを右へ
C-b または ←	カーソルを左へ
C-n または ↓	カーソルを下へ
C-p または ↑	カーソルを上へ
C-d	カーソル位置の文字消去
C-BackSpace	カーソル位置直前の文字消去
C-g	取り消し(画面が乱れた時に使用)
C-x C-s	ファイルにセーブ
C-x C-c	Emacs の終了

48

Emacsのコマンド(追加)

キー操作	意味
C-k	カーソルから右を切り取り
C-y	直前に切り取った部分を貼り付け
C-a	行の先頭に移動
C-e	行の末尾に移動

49

切り取りと貼り付けを利用した行のコピー

```
program hello
  write(*, *) 'All you need is love.'
  write(*, *) 'All you need is love.'
stop
end program
```

C-k C-k C-y C-y

```
write(*, *) 'All you need is love.'
改行
```

50

連続行のコピー

- 連続行の切り取り:
C-k を連続して押すと連続行を切り取り可
- C-k 以外の操作を行うと連続行の切り取り終了
 - C-y を押すとその時点で切り取られた連続行を全て貼り付け
 - C-k 以外の操作を行った後再び C-k を押すと、それまでの切り取り分はクリアされる。

51

日本語入力

- emacs を起動する前に

```
% setenv LANG ja_JP
```

- 手順

1. C-Y	モード行の左側が [あ] になる
2. ローマ字入力	入力文字の両端に が表示される
3. スペース	漢字変換開始
4. 正しい変換が行われるまで編集	
スペース	次候補
C-p	前候補
C-f	次の文節へ移動
C-b	前の文節へ移動
C-o	文節を大きくする
C-i	文節を小さくする
5. 改行	確定

- 初めて日本語を入力する際、辞書を作成するかどうかを
何回かたずねられるので、全て y を入力する。

52

Emacsの練習

- Windows等で利用するエディタとは
操作方法が大きく異なるので、
多少の練習が必要。
- チュートリアル:
Emacsの操作を練習するためのプログラム

C-h T

53

Emacsで変な画面になったときは

- 何かのコマンドを誤って入力した場合、
そのコマンドに関するメッセージが表示される。
- まず、C-g (取り消し) を数回押してみる。
 - 元に戻ったら、入力を続ける。
- 元に戻らなければ、一旦 C-x C-c で
Emacs を終了する。
 - Save file ? には y を入力。
 - その後、あらためて
% emacs さっき編集していたファイルの名前
でEmacsを起動。

54

スケジュール

- UNIXの概要
- 計算機への接続, ログイン, ログアウト
- ファイル操作
- テキストファイルの編集
- プログラムの翻訳, 実行
- その他のコマンド
- ウィンドウシステム (X Window)

55

プログラムの翻訳と実行

- プログラムのコンパイル(翻訳)とは?
 - 通常我々が作成するプログラムは「高級言語」(=人間向けの言葉)で書かれたプログラムであり, コンピュータはそのままでは実行できない.
 - コンピュータがそのまま実行できる「機械語」で書かれたプログラムは非常にわかりにくいため, 人間が作成, 編集するのは困難.

56

機械語と高級言語

- 例えば $5 + 7$ の計算
 - 高級言語の命令:
 $5 + 7$
 - 機械語の命令:
 - 場所A に 5を格納
 - 場所B に 7を格納
 - 場所A の値と場所B の値を加算
- まず, 分かりやすい言語(高級言語)で書いて, 機械語に翻訳する.
 - 高級言語の例: Fortran, C, Pascal, Java, Basic等

57

コンパイラ

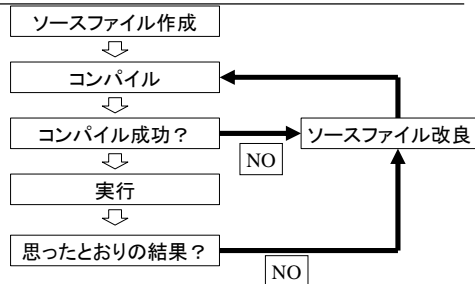
- 高級言語を機械語に翻訳するソフトウェア



- ソースファイル: 高級言語で書かれたプログラム
- 実行ファイル: コンパイラによって生成される機械語で書かれたプログラム

58

プログラム開発の流れ



- ソースファイルを書き換えたら必ずコンパイル

59

Fortranプログラムのコンパイル f90 もしくは frt コマンド

- 使い方:
`f90 ソースファイル -o 実行ファイル`
- 例: hello.f90 をコンパイルして実行ファイル hello を作成
`% f90 hello.f90 -o hello`
- FORTRAN77 のコンパイルは f77 コマンド
- 富士通の計算機(kyu-vpp)では全て frt コマンド

60

C言語プログラムのコンパイル cc コマンド

- 使い方:
cc ソースファイル -o 実行ファイル
- 例: hello.c をコンパイルして実行ファイル hello を作成
% `cc hello.c -o hello`

61

ファイルの拡張子

- コンパイラはファイルの拡張子(ファイル名の末尾の文字列)でファイルの形式を判断する.

- Fortran

.f	FORTRAN77
.f90	Fortran 90
.f95	Fortran 95

- C 言語

.c	C 言語
.C	C++ 言語

62

プログラムの実行

- コンパイルによって得られた実行ファイルの名前をコマンドとして利用する.
- 例: カレントディレクトリの実行ファイル hello の実行
% `./hello`
- UNIXのコマンドと区別するために、実行ファイルの前に必ず `./` を付ける.

63

コンパイル時の最適化オプション

- プログラムの実行に時間がかかる場合、以下のオプションを用いてもう一度コンパイルし、実行してみる.

-O	基本的な最適化
-O3	より高度な最適化(演算評価方法等)
-O4	-O3よりさらに高度な最適化
-O5	最大限の最適化

- 例)

```
kyu-cc% cc -O3 mm.c -o mm
```

(詳細は「プログラミング言語講習会」で)

64

スケジュール

- UNIXの概要
- 計算機への接続, ログイン, ログアウト
- ファイル操作
- テキストファイルの編集
- プログラムの翻訳, 実行
- その他のコマンド
- ウィンドウシステム(X Window)

65

実行結果をファイルに出力, 実行時にデータをファイルから入力

- リダイレクション
 - > 左側のコマンドからの出力を画面に表示せず右側のファイルに保存
 - >> 左側のコマンドからの出力を画面に表示せず右側のファイルに追加
 - < 左側のコマンドに対する入力としてキーボードの代わりに右側のファイルの内容を利用

例) `ls` コマンドの結果を整理して `rdirs` に保存

```
kyu-cc% ls > dirs
kyu-cc% sort -r < dirs > rdirs
```

```
hello.c
hello2.c
work/
```

```
work/
hello2.c
hello.c
```

66

実行結果を他のコマンドの入力に利用

- パイプライン
 - | 左側のコマンドからの出力を画面に表示せず右側のコマンドの入力として利用

例) ls コマンドの結果を整理して rdirs に保存

```
kyu-cc% ls | sort -r > rdirs
```

```
work/  
hello2.c  
hello.c
```

67

実習2 プログラムの編集, コンパイル, 実行

- Emacs を使ったプログラムの編集
- コンパイル
- 実行
- リダイレクトとパイプライン

68

プリンタ

- プリンタへの出力
 - lp プリンタへ出力 (情報基盤センター2F)

- 例1: テキストファイル
 - % lp hello.txt
- 例2: PostScriptファイル
 - % lp hello.ps

69

オンラインマニュアルの利用

- 使い方:
 - man 調べたいコマンド

- 例: コマンド ls のマニュアルを表示
 - % man ls

```
LS(1)      ls (fileutils) 4.1 (April 2001)    LS(1)  
NAME  
ls - list directory contents  
SYNOPSIS  
ls [OPTION]... [FILE]..  
DESCRIPTION  
List information about the FILES (the current directory by  
default). Sort entries alphabetically if none of -ftuSUX  
nor --sort.  
-a, --all  
do not hide entries starting with .  
-A, --almost-all  
do not list implied . and ..  
-b, --escape  
print octal escapes for nongraphic characters
```

70

漢字コード変換

- 使い方
 - nkf オプション ファイル

- 計算機内部では文字が番号(コード)で扱われる.
 - 日本語(漢字, ひらがな, カタカナ)のコードは3種類
 - EUC 主に UNIX で利用
 - JIS 主にインターネットで利用
 - Shift-JIS 主にPC(Windows, Macintosh)で利用
 - 改行コードも UNIX, Windows, Macintosh で異なる
- 表示や動作が異常なプログラムは変換してみる.
- 例: 日本語コードと改行コードをUNIX 向けに変換

```
kyu-cc% nkf --euc --unix hello.c > hello-e.c
```

71

センターの計算機に関するコマンド

- ワークステーションへの登録
 - touroku 当センター研究用システムの
 - 並列計算サーバ kyu-ss
 - プリポスト処理用ワークステーション vhsunに対して利用登録する
(kyu-vpp.cc.kyushu-u.ac.jp で利用可能)
- 課金情報の表示
 - ucharge 現在の課金情報を表示

```
=====
```

DATE	USER	CHARGE	TOTAL	TAX	PRETAX
*2005.03.**	k70043a	61,551	1,069,498	3,077	64,628
>2005.04.27	k70043a	63,782	66,971	3,189	66,971

```
=====
```

スケジュール

- UNIXの概要
- 計算機への接続, ログイン, ログアウト
- ファイル操作
- テキストファイルの編集
- プログラムの翻訳, 実行
- その他のコマンド
- ウィンドウシステム (X Window)

73

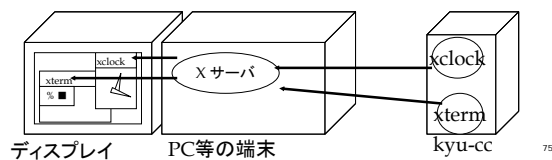
X-Window システム

- UNIX で複数の画面(ウィンドウ)を同時に表示、操作させるためのシステム
 - 基本的な操作方法は Macintosh, Windows 等と同じ
 - マウスによる操作 (クリック, ドラッグアンドドロップ等)
- X サーバソフトの利用により Windows 上でも X Window を起動できる
 - Astec-X, Exceed, Cameleon等

74

X サーバと X クライアント

- X サーバ:
 - 画面の表示を制御するプログラム
 - 各ディスプレイに一つ
 - X クライアントからの要求に応じて画面表示
- X クライアント
 - 画面への表示を要求するプログラム
 - xclock, kterm, xcalc 等



75

Xサーバソフト (Astec-X) で接続

- まず TeraTerm等で kyu-cc にログインしておく。
- Windows でスタートボタンをクリックし、[プログラム]→[ASTEC-X]→[ASTEC-X]でタスクバー(画面右下)に [X]のアイコンが表示される。
- TeraTerm のウィンドウで以下を実行する。

```
kyu-cc% setenv DISPLAY 133.5.11.???:0.0
Kyu-cc% xterm &
Kyu-cc%
```

ディスプレイ右下の 133.5.11.??? の番号

バックグラウンド

- コマンド入力と平行してプログラムを実行
 - 見かけ上、同時に複数の処理を実行

通常:

```
kyu-cc% xterm
```

バックグラウンド: コマンドの後ろに & を付ける

```
kyu-cc% xterm &
kyu-cc%
```

注意:

ログアウトしてもバックグラウンドジョブが残ることがあるので、全てのバックグラウンドジョブを終了させてからログアウトする

お疲れ様でした

- 実習で利用した ID は1週間利用できます。
- 「X-Window」に関する実習は4階の「第一自習室」で試すことができます。
- 不明な点等は
request@cc.kyushu-u.ac.jp
もしくは
nanri@cc.kyushu-u.ac.jp
まで。

78