コンピュータリテラシー

第4回　コンピューティングの要素と構成

平成22年5月10日(月)

第4回 コンピューティングの要素と構成 2/40

計算機の基本構成、値の表現と処理。

1. 前回の情報倫理演習1について

   次回に相互評価を行う。
   グループごとにWikiのページを完成させておくこと。

   入力で改行がうまくいかないときは、ブラウザにFirefoxを使う。
   Safariでは改行が入らないようである。
   FirefoxはFinderのアプリケーションから選ぶ。
   ドックに格納しておくと便利である。
   ドックに格納するには、Firefoxのアイコンをドラッグして、ドックで離す。

   情報の検索のために「WWWでの検索」を自習すること。

2. 「(4.1)計算機内部でのデータの表現」と「(4.2)計算機の構造」について学ぶ。
3. 「(4.3)Unix の基本事項」について復習しておくこと。
4. 「(4.4)ファイルシステム入門」について復習しておくこと。

まず、今日やること(使い勝手)

- ブラウザーの選択
  「Safari」or「Firefox」

- コンピュータリテラシーのホームページ:
  http://172.21.54.48/moodle/
  を、ブックマーク（お気に入り）に登録しよう。

第4回 コンピューティングの要素と構成

4.1「計算機内部でのデータの表現」
4.2「計算機の構造」
4.3「Unix の基本事項」(コマンド)
4.4「ファイルシステム入門」
　　(ディレクトリ と ファイル)

第4回 コンピューティングの要素と構成

4.1「計算機内部でのデータの表現」

4.2「計算機の構造」

4.3「Unix の基本事項」(コマンド)

4.4「ファイルシステム入門」
(ディレクトリ と ファイル)

説明のなかった項目の学習と演習を行う。

4.1 計算機内部でのデータの表現

4.1.1　2進数, 8進数, 16進数
4.1.2　「文字」の扱い (英字)
4.1.3　「文字」の扱い (日本語)
4.1.4　日本語の文字コードの変換
4.1.5　「整数」の表現　「2の補数表現」
4.1.6　小数点のついた数

「基数」について

- たとえば、3桁の10進数「１８２」という数は、

$$182 = 100 + 80 + 2$$

$$= (1 \times 100) + (8 \times 10) + (2 \times 1)$$

$$= (1 \times 10^2) + (8 \times 10^1) + (2 \times 10^0)$$

と表現できる。

- 10進法における「１０」を「基数」という。
- 各桁の数字は、0～9までの10個を利用できる。

2進数(binary number)

- 基数は「２」
- 「０」と「１」の2個の数字のみで数を表現する。
- 3桁の２進数「110」という数を10進数に変換すると

$$(1\times 2^2)+(1\times 2^1)+(0\times 2^0)$$
$$=(1\times 4)+(1\times 2)+(0\times 1)$$
$$=4+2+0$$
$$=6$$

3桁の10進数「182」を2進数に変換すると
8桁の　2進数「10110110」となる:

$$182 \div 2 = 91 \cdots 0$$

$$91 \div 2 = 45 \cdots 1$$

$$45 \div 2 = 22 \cdots 1$$

$$22 \div 2 = 11 \cdots 0$$

$$11 \div 2 = 5 \cdots 1$$

$$5 \div 2 = 2 \cdots 1$$

$$2 \div 2 = 1 \cdots 0$$

$$1 \div 2 = 0 \cdots 1$$

ビット(bit)とバイト(byte)

- 0と1のどちらかを区別する情報の最小単位を「ビット」 という.
- つまり、2進数の一桁(の単位)を「ビット」という

  3桁の10進数「182」

  8桁の 2進数「10110110」

- 「バイト」とは、8ビットを単位として扱うこと。
- つまり、「1バイト」＝「8ビット」

8進数(octal number) : 基数は「8」

- 「0～7」の8個の数字で数を表現
- 8個の数字を表現するのに必要なビット数は3.

$$000 \Leftrightarrow 0$$

$$001 \Leftrightarrow 1$$

$$010 \Leftrightarrow 2$$

$$\vdots \Leftrightarrow \vdots$$

$$111 \Leftrightarrow 7$$

16進数(hexadecimal number)：基数は「１６」

- 10個の数字「0～９」と6個の文字「a,b,c,d,e,f」の合わせて16個で数を表現（a=10,b=11,…,f=15）
- 16個の数字を表現するのに必要なビット数は4.

$$0000 \Leftrightarrow 0$$

$$0001 \Leftrightarrow 1$$

$$0010 \Leftrightarrow 2$$

$$\vdots \Leftrightarrow \vdots$$

$$1110 \Leftrightarrow 14 = \mathrm{e}$$

$$1111 \Leftrightarrow 15 = \mathrm{f}$$

それぞれの関係(10進数「182」)

- 「10110110」:2進数(8桁)
- 「266」: 8進数(3桁)
- 「182」: 10進数（3桁）
- 「b6」:16進数(2桁)

- 「10110110」⇒「 10,110,110」⇒「2,6,6」(8進数)
- 「10110110」⇒「 1011, 0110」⇒「11,6」
  ⇒「b,6」(16進数)

4.1 計算機内部でのデータの表現

4.1.1　2進数, 8進数, 16進数

4.1.2　「文字」の扱い (英数字と記号)

4.1.3　「文字」の扱い (日本語)

4.1.4　日本語の文字コードの変換

4.1.5　「整数」の表現　「2の補数表現」

4.1.6　小数点のついた数

英数字と記号(1バイトの文字コード)

- ASCII (アスキー)
- American Standard Code for Information Interchange
- 8ビット（1バイト）で表現
  (なぜなら、「英数字と記号」は全部で95種類)
- たとえば、アルファベットの大文字「A」:
- 「A」⇔「01000001」(2進数)
- 「A」⇔「４１」(16進数)「0100,0001」⇔「4,1」
- 「A」⇔「６５」(10進数)

確認方法(後で確認)

- 「A」⇔「01000001」(2進数)
- 「A」⇔「41」(16進数)⇔「65」(10進数)
- 「a」⇔「01100001」(2進数)
- 「a」⇔「61」(16進数)⇔「97」(10進数)

- 確認方法: コマンド od を利用する:

〉od –t x1 foo.txt (16進数)

〉od –t d1 foo.txt (10進数)

4.1 計算機内部でのデータの表現

4.1.1　2進数, 8進数, 16進数

4.1.2　「文字」の扱い (英数字と記号)

4.1.3　「文字」の扱い (日本語)

4.1.4　日本語の文字コードの変換

4.1.5　「整数」の表現　「2の補数表現」

4.1.6　小数点のついた数

日本語(2バイトの文字コード)

- JIS
  メールの配送などに使われている.
- MS漢字
  マイクロソフト社が定めたコード体系.
  「Shift JIS」 とか 「SJIS」 ともいう.
  Windows や MacOS が標準で使っている.
- EUC
  おもに Unix 系で使われている.
- UTF−8
  ユニコードともいう. もっとも新しく制定されたコード体系.

日本語の文字コードの変換(後で確認)

- コマンド　nkf　を利用する。

〉nkf –j foo.txt　(JISへ変換)

〉nkf –s foo.txt　(SJISへ変換)

〉nkf –e foo.txt　(EUCへ変換)

〉nkf –w foo.txt　(UTF-8へ変換)

nkf は Nihongo Kanji Filter の略.

4.1 計算機内部でのデータの表現

4.1.1　2進数, 8進数, 16進数

4.1.2　「文字」の扱い (英数字と記号)

4.1.3　「文字」の扱い (日本語)

4.1.4　日本語の文字コードの変換

4.1.5　「整数」の表現　「2の補数表現」

4.1.6　小数点のついた数

整数の表現

- 4桁の2進数 $b_3b_2b_1b_0$ を考える
- 最上位ビット: b_3 ⇔ 最下位ビット: b_0

- 4ビット $b_3b_2b_1b_0$ を10進数に変換するには、

$$(b_3 \times 2^3) + (b_2 \times 2^2) + (b_1 \times 2^1) + (b_0 \times 2^0)$$

- これにより、4ビットの場合、0～15までの16通りの非負の整数を表現できる。
- つまり、. . .

$0000 \Leftrightarrow 0$

$0001 \Leftrightarrow 1$

$0010 \Leftrightarrow 2$

$0011 \Leftrightarrow 3$

$0100 \Leftrightarrow 4$

$0101 \Leftrightarrow 5$

$0110 \Leftrightarrow 6$

$0111 \Leftrightarrow 7$

$1000 \Leftrightarrow 8$

$1001 \Leftrightarrow 9$

$1010 \Leftrightarrow 10$

$1011 \Leftrightarrow 11$

$1100 \Leftrightarrow 12$

$1101 \Leftrightarrow 13$

$1110 \Leftrightarrow 14$

$1111 \Leftrightarrow 15$

加法と減算について

- ４＋２＝６

$$\begin{array}{rcl} & 0\ 1\ 0\ 0 & \Leftrightarrow\ 4 \\ + & 0\ 0\ 1\ 0 & \Leftrightarrow\ 2 \\ \hline & 0\ 1\ 1\ 0 & \Leftrightarrow\ 6 \end{array}$$

- ６－２＝４

$$\begin{array}{rcl} & 0\ 1\ 1\ 0 & \Leftrightarrow\ 6 \\ - & 0\ 0\ 1\ 0 & \Leftrightarrow\ 2 \\ \hline & 0\ 1\ 0\ 0 & \Leftrightarrow\ 4 \end{array}$$

2の補数表現

- 4桁の2進数　$b_3 b_2 b_1 b_0$を考える
- 「2の補数表現」では、最上位ビットb_3の項をマイナス項として、10進数に変換する。すなわち、

$$-(b_3 \times 2^3) + (b_2 \times 2^2) + (b_1 \times 2^1) + (b_0 \times 2^0)$$

- これにより、4ビットの場合、－8 ～ 7　までの16通りの整数を表現できる。
- つまり、. . .

$0000 \Leftrightarrow 0$

$0001 \Leftrightarrow 1$

$0010 \Leftrightarrow 2$

$0011 \Leftrightarrow 3$

$0100 \Leftrightarrow 4$

$0101 \Leftrightarrow 5$

$0110 \Leftrightarrow 6$

$0111 \Leftrightarrow 7$

$1000 \Leftrightarrow -8$

$1001 \Leftrightarrow -7$

$1010 \Leftrightarrow -6$

$1011 \Leftrightarrow -5$

$1100 \Leftrightarrow -4$

$1101 \Leftrightarrow -3$

$1110 \Leftrightarrow -2$

$1111 \Leftrightarrow -1$

加法と減算について(2の補数表現)

- ４＋２＝６

$$\begin{array}{rcl} 0\ 1\ 0\ 0 & \Leftrightarrow & 4 \\ +\ 0\ 0\ 1\ 0 & \Leftrightarrow & 2 \\ \hline 0\ 1\ 1\ 0 & \Leftrightarrow & 6 \end{array}$$

- ６－２＝６＋（－２）＝４

$$\begin{array}{rcl} 0\ 1\ 1\ 0 & \Leftrightarrow & 6 \\ +\ 1\ 1\ 1\ 0 & \Leftrightarrow & -2 \\ \hline 1\ 0\ 1\ 0\ 0 & \Leftrightarrow & 4 \end{array}$$

- 4ビットのみに着目する。（5ビット目を無視）

「補数」について

- たとえば、2の補数表現において「3」と「－3」は:
- 「　3」⇔「0011」
- 「－3」⇔「1101」
- 3＋(－3)＝0

$$\begin{array}{rcl} 0\ 0\ 1\ 1 & \Leftrightarrow & 3 \\ +\ 1\ 1\ 0\ 1 & \Leftrightarrow & -3 \\ \hline 1\ 0\ 0\ 0\ 0 & \Leftrightarrow & 0 \end{array}$$

- 5ビット目のみが「1」で、それ以外は「0」となる

4.2 計算機の構造(各自で学習)

- 計算機にはどのような処理ができるか

4.2.1　計算機の構成要素
4.2.2　計算機の動作
4.2.3　プログラム言語
4.2.4　オペレーティングシステム(OS)

第4回 コンピューティングの要素と構成

4.1「計算機内部でのデータの表現」

4.2「計算機の構造」

4.3「Unix の基本事項」(コマンド)

4.4「ファイルシステム入門」

(ディレクトリ と ファイル)

4.3 Unix の基本事項

1. Unix「コマンド」の書式
2. 「オンラインマニュアル」の参照
   man コマンド
3. ファイルの「一覧表示」
   ls コマンド (LとS)
4. ファイルの「内容の表示」
   cat コマンド
   more コマンド

第4回 コンピューティングの要素と構成

4.1「計算機内部でのデータの表現」

4.2「計算機の構造」

4.3「Unix の基本事項」(コマンド)

4.4「ファイルシステム入門」
(ディレクトリ と ファイル)

Unix「コマンド」の書式

〉コマンド □［オプション］□［引数］

例えば、ファイルの複製(コピー, copy)を作成するコマンドは　cpコマンド　である。次のように実行する。

〉cp　－i　foo.txt　goo.txt

ファイル名　foo.txt の複製を別のファイル名 goo.txt として作成する。

それでは、実際にコマンドを実行してみましょう。

第4回 コンピューティングの要素と構成

4.1「計算機内部でのデータの表現」

4.2「計算機の構造」

4.3「Unix の基本事項」(コマンド)

4.4「ファイルシステム入門」
(ディレクトリ と ファイル)

4.4 「ファイルシステム入門」

1. ls コマンド〔一覧表示〕
   (先ほどのUNIXの基本事項でも扱った)
2. pwd コマンド〔ホームディレクトリの確認〕

それでは、実際にコマンドを実行してみましょう。

以上

1 Emacsの復習

以下を第2回教材 を参照しながら、順に行うこと。

1. キーボード

を見ずに、左手人差し指でキーF上の突起を手がかりにキーFを探り、その上に人差し指を置く。左手の中指、薬指、小指を順にキーD、キーS、キーA上に置く。
右手も同様に、キーボードを見ずに、人差し指でキーJ上の突起を手がかりにキーJを探し、その上に人差し指を置く。右手の中指、薬指、小指を順にキーK、キーL、キー;上に置く。
これをホームポジションという。手をキーボードから離し、キーボードを見ずにホームポジションに手を置くことを何度か行ってみよう。

2. メタキー、コントロールキーはそれぞれキーボード上のどこにあるか？ Emacs のコマンドの表記
   タブキー(Tab)、エスケープキー(Esc)の位置も確認しておこう。
3. kterm に表示されている(シェル)プロンプトを確認せよ。
   また、> ls を実行してみよう。
4. kterm上で、オンラインマニュアルを用いてコマンド ls のコマンド表記の形式を調べよ。
   コマンド ls のコマンド引数・オプションにどのような記述ができるか。

ヒント

5. kterm上で、アクセス権やファイルの大きさなどを含む
   詳細なファイルの一覧を表示してみよう。(シェル)コマンド

ヒント

6. emacsを起動し、
   ホームポジションに手を置き、タッチタイプの練習をしてみよう。
   emacsを終了するコマンドは何か？

ヒント

7. エディタemacsを用いて
   Hello World!
   という文字列を1行含む hello.txt という名前のファイルを作成しよう。
   ファイルの操作 (1)

ヒント

8. もう一度、エディタemacsを用いて hello.txt を編集し、もう1行、適当な文を追加しよう。
   Emacs の基本編集コマンド (1)
   1. テキストの入力と挿入
   2. カーソルの移動 (1)
   3. テキストの削除・コピー・移動 (1)
   4. テキストの探索
   5. とりやめと取消し

ヒント

   Emacsの起動と終了 (2)参照。
9. カレントディレクトリがどこか確認しよう。(カレントディレクトリ名の表示)

ヒント

10. カレントディレクトリをホームディレクトリへ変更しよう。(カレントディレクトリの変更, ホームディレクトリ)

40/40

ヒント

11. ホームディレクトリにあるファイル名を表示してみよう。(ホームディレクトリ)

ヒント

12. ホームディレクトリ中のディレクトリmacがあり、
そのディレクトリmacの中に cl という名のディレクトリがありますか？
(授業に関連した環境設定の確認)
無ければ授業に必要な環境設定を行い、
cl という名のディレクトリの存在を確認してください。。

ヒント

13. ホームディレクトリに、自分の学籍番号 氏名
の1行からなるexercise03.txt という名のファイルをemacsを用いて作成しよう。(eggでの日本語入力)

ヒント