情報処理 (2)

日本語の入力・エディタ・電子メール

増原 英彦

1 エディタによるテキストファイルの編集

練習 2–1: (エディタの基本操作) HWB の「15. エディタ」を読み、エディタの基本操作を習得せよ。HWB では「テキストエディット編」と「Emacs 編」という 2 種類のエディタが紹介されているが、この授業では Emacs をすすめる。今後の実習を続けるためには、最低でも以下のような操作が必要となる:

- ファイルを開く・ファイルへ保存・文字の挿入・削除
- 文章の切り貼り・検索・置換
- 1 つのファイルの文章を (全て) 別のファイルに写す

練習 2–2: (文字コードの確認)** ASCII 符号化されたファイルの文字のコードを次のようにして調べよ。

1. まず、エディタを用いて英語アルファベットと数字だけを用いた適当な文章を書いたファイル `ascii.text` を作成する。
2. 「ターミナル」を起動 (HWB12.2.1) し、上で保存したファイルが存在するディレクトリに移動 (HWB13.4.3) し、`hexdump -C ascii.text` のようなコマンドを実行して各ファイルの内容を 16 進数で表示させよ。
3. 授業スライドにあった ASCII コード表から、文字コードとファイルの内容が一致することを確めよ。

2 日本語の入力

練習 2–3: (入力の方法) HWB 9 章「日本語の扱い」を読み、日本語を入力するための基本的な方法を習得せよ。

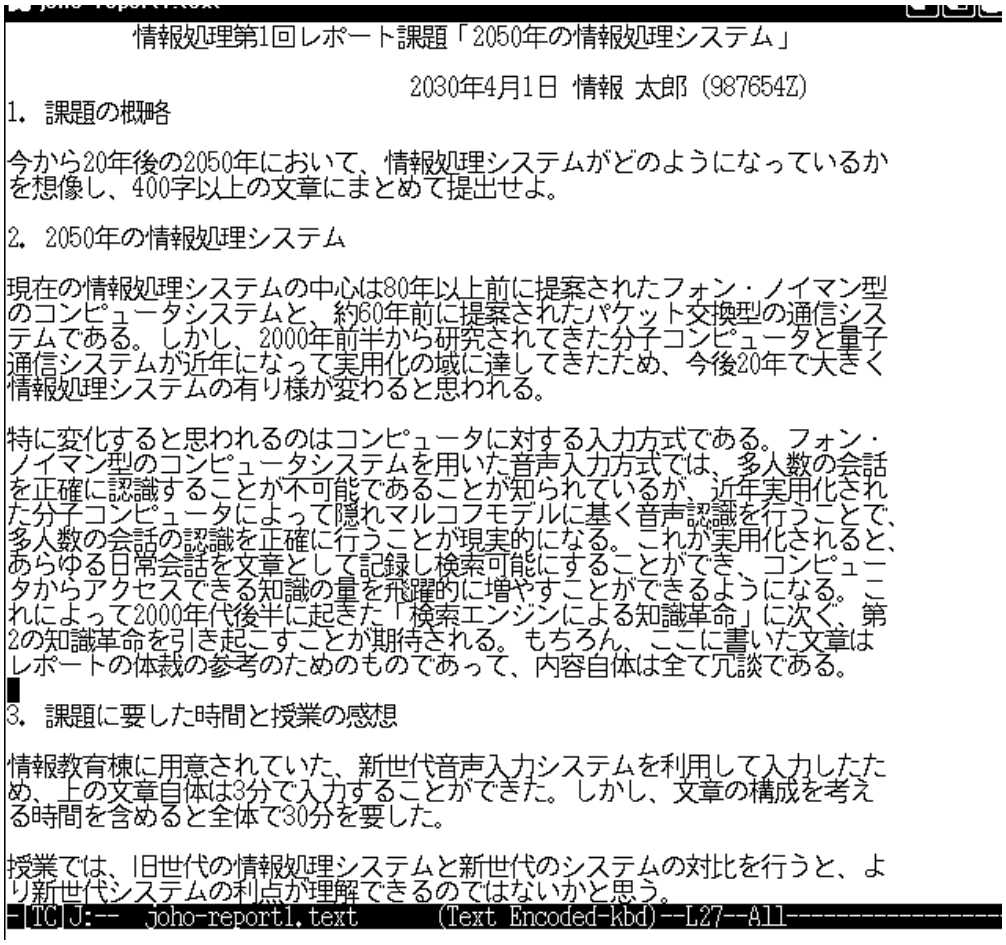
- この授業では、基本的な文章が入力できれば問題ない。
- 自分好みの入力方法でないと感じる場合は、入力状態メニューの「環境設定を表示」を選ぶと、ある程度の変更ができる。

練習 2–4: (日本語の編集) 次のような題で 400 字以上の文章を書き、図 1 のような形式でファイルに保存せよ。

題: コンピュータネットワークにおける匿名の情報発信は守られるべきか?

(現在、匿名掲示板やファイル交換システムのような情報システムの中には、発信者の身元を明らかにしないでも情報発信が可能なものがある。このようなシステムに対して、違法行為を防ぐために匿名の情報発信を規則する動きと、通信や表現の自由のためには匿名性が不可欠であるという立場がある。自分がどちらかの立場をとるかを明記した上で、情報発信の匿名性に対する意見を書け。)

なお、示したものはレポートの一例であって、必ずしもこの通りに作る必要はない。



情報処理第1回レポート課題「2050年の情報処理システム」

2030年4月1日 情報 太郎 (987654Z)

1. 課題の概略

今から20年後の2050年において、情報処理システムがどのようになっているかを想像し、400字以上の文章にまとめて提出せよ。

2. 2050年の情報処理システム

現在の情報処理システムの中心は80年以上前に提案されたフォン・ノイマン型のコンピュータシステムと、約60年前に提案されたパケット交換型の通信システムである。しかし、2000年前半から研究されてきた分子コンピュータと量子通信システムが近年になって実用化の域に達してきたため、今後20年で大きく情報処理システムの有り様が変わると思われる。

特に変化すると思われるのはコンピュータに対する入力方式である。フォン・ノイマン型のコンピュータシステムを用いた音声入力方式では、多人数の会話を正確に認識することが不可能であることが知られているが、近年実用化された分子コンピュータによって隠れマルコフモデルに基く音声認識を行うことで、多人数の会話の認識を正確に行うことが現実的になる。これが実用化されると、あらゆる日常会話を文章として記録し検索可能にすることができ、コンピュータからアクセスできる知識の量を飛躍的に増やすことができるようになる。これによって2000年代後半に起きた「検索エンジンによる知識革命」に次ぐ、第2の知識革命を引き起こすことが期待される。もちろん、ここに書いた文章はレポートの体裁の参考のためのものであって、内容自体は全て冗談である。

3. 課題に要した時間と授業の感想

情報教育棟に用意されていた、新世代音声入力システムを利用して入力したため、上の文章自体は3分で入力することができた。しかし、文章の構成を考える時間を含めると全体で30分を要した。

授業では、旧世代の情報処理システムと新世代のシステムの対比を行うと、より新世代システムの利点が理解できるのではないかと思う。

-[TC]J:-- joho-report1.text (Text Encoded-kbd)--L27--All----

図 1: レポートの形式の例

練習 2–5: (漢字コードの調査)** 自分の氏名の漢字コードと、それがどのように符号化されるかを次のようにして調べよ。

1. まず、HWB 15.3.2.7 を参考にして、自分の名前が書かれたファイルを作り、異なる名前で保存せよ。例えば、以下のようなファイル名と文字コードで保存してみよ。

ファイル名	文字コードの名前	符号化方式
myname-utf8.text	utf-8	UTF-8
myname-utf16.text	utf-16	UTF-16
myname-jis.text	iso-2022-jp	JIS
myname-sjis.text	sjis	シフト JIS
myname-euc.text	euc-jp	日本語 EUC

2. 「ターミナル」を起動 (HWB12.2.1) し、上で保存したファイルが存在するディレクトリに移動 (HWB13.4.3) し、`hexdump -C myname-utf8.text` のようなコマンドを実行して各ファイルの内容を 16 進数で表示させよ。

3. 入力状態アイコン (HWB9.1.1) をクリックし、「“文字パッレト” を表示」を選び、文字パレットを表示させよ。文字パレットの中から、氏名の文字を選択すると、その文字コードが表示される。表示された文字コードと 2 で表示されたファイルの内容がどのように対応しているかを推測せよ。

3 電子メール

練習 2–6: (シミュレータによる電子メールの原理)　HWB「8.4.5. 電子メールシミュレータ」に従って、電子メールがどのように送られるかを実験し、以下の疑問に答えよ。

1. 100 人に同じ内容のメールを一斉に送る場合、どのコンピュータの間では何回通信が行われるだろうか? user3@host.b.jp から

   (a) user1@host.a.jp と user4@host.c.jp に一つのメッセージを一斉に送った場合と、

   (b) user1@host.a.jp と user2@host.a.jp に一つのメッセージを一斉に送った場合で

   ネットワーク上を流れるメッセージの数を観察して推測せよ。

2. IMAP サーバと POP サーバでは、利用者がメールを読んだ際にどのような違いがあるか? user1@host.a.jp と user4@host.c.jp にメッセージを送り、メールを受信した場合にそれぞれのメールサーバのスプールがどのように変わるかを観察して推測せよ。

3. 誤りのあるメールアドレスにメッセージを送った場合、どこでエラーが起きるか? ドメイン名とユーザ名のそれぞれに誤りがある場合についてメッセージを送り、その様子を観察して推測せよ。

練習 2–7: (自分自身への送受信)

1. HWB 「8.1 電子メール」「8.2.1 Apple Mail の起動と終了」「8.2.2 メッセージを送る」を読み、自分自身にメッセージを一通送れ。件名は「練習 2–7 」とせよ。内容は何でもよいが何か書いて送ってみよ。

2. 「8.2.3 メッセージを読む」で、自分自身に送られたメッセージを読んでみよ。

3. 「8.2.4 届いたメッセージに返信する」に従って、自分自身に返信を送ってみよ。

練習 2–8: (エラーメールの読み方)

1. 間違ったドメイン名のメールアドレスにメッセージを送り、どのような返事が来るかを調べよ。例えば、[自分のログイン名]`@mail.ecc.u-tokyo.ac.japan` というアドレスに送ってみよ。

2. 間違ったユーザ名のメールアドレスにメッセージを送り、どのような返事が来るかを調べよ。例えば、`g98765432@mail.ecc.u-tokyo.ac.jp` というアドレスに送ってみよ。エラーメッセージは 1 とどう違うか?

練習 2–9: (実際に転送されるデータ*)　上の練習で受信したメッセージが、実際にどのようなデータであるかを観察せよ。「Mail」(バージョン 1.3.7) でメッセージの実際の内容を見るためには、メッセージを表示し、メニューの [表示] → [メッセージ] → [ソース] を選べばよい。特に以下の点がどうなっているかを表示から推測せよ:

- メッセージが送られるまでに経由した計算機の名前
- メッセージの符号化方式はどのように指定されているか
- 日本語の「件名」はどのように符号化されているか
- 日本語の本文はどのように符号化されているか
- 「リッチテキスト」形式の本文はどのように符号化されているか
- 添付ファイルはどのように符号化されているか

練習 2–10: (メールの転送経路)** 電子メールが送られる際には、複数のメールサーバを経由してメッセージが届くことが多い。このことを、次のようにして確めよ。

1. 携帯電話等の学外のシステムから、学内の自分のアドレスにメッセージを送る。
2. メッセージを受信したら、メニューの 表示 → メッセージ → すべてのヘッダ (あるいは ソース) を選び、詳細な情報を表示させる。
3. `Received:` と書かれた行は、通常、メールサーバを通過する度に加えられる。練習 2–7 で送ったメッセージの場合と比べて、`Received:` と書かれた行がどのように違うかを確認せよ。また、ヘッダに書かれている時刻から、メッセージが送られてから受信されるまでにどのくらいの時間が経過しているかを確認せよ。

4 課題

課題 2–1: (エディタと電子メール) 練習 2–4 で作成したレポートを、提出用メールアドレス`masuhara-js-report@lecture.ecc.u-tokyo.ac.jp` および授業 WWW ページで指示された 2 名の履習者、および自分自身に同時に送れ。ただし、

- レポートは本文中に直接テキスト形式で書き込め。添付ファイルや、リッチテキスト形式は用いないこと。
- レポートには、課題を完了するのに要した時間と授業に対する感想も書くこと
- メールは必ず教育用計算機システムから送ること
- 自分自身にも送るのを忘れないこと
- 課題以外の「練習」を同時に提出してもよい

提出期限 2004 年 5 月 13 日 (金) 23 時 59 分