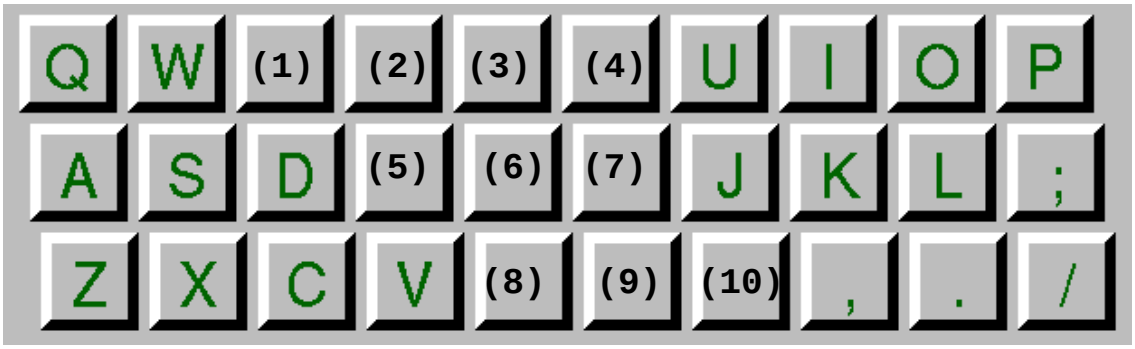
情報処理 試験問題 (田中哲朗)

2001 年 7 月 30 日 2 限 理科一類 22, 23 組 理科二類 / 理科三類 11 組

試験時間 30 分，解答用紙 両面 1 枚，持ち込み不可

第 1 問 (10 点)

下図は, 通常使われるキーボードの配列の一部を示すが，アルファベット 10 文字 (B, E, F, G, H, M, N, R, T, Y) が抜けている. 図中の空白 (1)-(10) がそれぞれ, どの文字に対応するかを答えなさい.



第 2 問 (15 点)

Goo や Google などのロボット型検索エンジンで検索をして，出てきたページを見たら全く関係のないページだということはよくある．どのような場合にこのようなことがおこるのか，ロボット型検索エンジンの原理を説明した上で，なるべく多くの可能性をあげ，箇条書きにしなさい (200 文字程度)．

第 3 問 (15 点)

以下の用語に関して 50 ~ 100 文字程度で解説を書きなさい．

- メールアドレス
- SPAM メール
- チェインメール

第 4 問 (15 点)

以下の文の空白部分 (a-e) を埋めなさい.

> Unix でハードディスク等に保存されるデータは, ファイル (file) と呼ばれるまとまりを単位として扱われる．多くのファイルを扱うので、いくつかのファイルを ディレクトリ (directory) と呼ばれる仮想的な 「(机の) 引き出し」 にまとめることができるようになっている．
>
> ディレクトリを作成するには [(a)] (make directory) というコマンドを用いる．コマンドラインから，
>
> [(a)] sub1
>
> と入力すると，sub1 というディレクトリができる．
>
> ディレクトリの中に更にディレクトリを作って，階層化することもできる．上で作ったディレクトリ sub1 の下の階層に sub-child というディレクトリを作るには，コマンドラインから，
>
> [(a)] [(b)]
>
> と入力すると作ることができる．そのディレクトリの下に Happy.txt というファイルを作るには，

```
echo 'I am happy.' > sub1/sub-child/Happy.txt
```

のようにする．ディレクトリは，親ディレクトリが 1 つで子ディレクトリを複数持つことができるという木 (Tree) 構造をしている．親ディレクトリを下にして絵を描いた時の一番下にあたるディレクトリが Unix システム内には一つだけ存在し，ルートディレクトリ (root directory) と呼ぶ．

参照の基準としているディレクトリを、 カレント・ディレクトリ (current directory) と呼ぶ． cd(change directory) というコマンドを使うと，カレント・ディレクトリを変えることができる．たとえば，

```
cd sub1
```

を実行すると，先ほど作った Happy.txt は，[(c)]という名前で参照できるようになり，

```
cat [ (c) ]
```

とすると，ファイルの内容を表示することができる．

ファイルやディレクトリを指定するには，カレント・ディレクトリ を基準にした相対パス (path) 指定と絶対パス指定の 2 種類の方法がある．たとえば，

```
cd /tmp
[ (a) ] sub1
[ (a) ] sub2
echo 'I am happy.' > sub1/happy.txt
cd sub2
```

としてディレクトリを 2 つ，ファイルを 1 つ作ったあとで，作ったファイルを表示するには相対パス指定で

```
cat [ (d) ]
```

とすることもできるし，，絶対パスで，

```
cat [ (e) ]
```

と指定することもできる．

第 5 問 (15 点)

以下の文の空白部分 (a-e) を埋めなさい.

計算機で画像を表現する方法としては，主に

- [(a)]表現
- [(b)]表現

の 2 種類がある．[(a)]表現は点、直線、円、などの幾何的な記述によって図形を表現するので，縮小や拡大などの変換を行う際に情報が失われることが[(c)]．

[(b)]表現は画像を規則的な 格子に分割して、画素 (ピクセル) と呼ばれる最小単位を並べることによって画像を表現する．

各ピクセルには色情報が与えられるが，色の表現の各ピクセルに **RGB (Red Green Blue)** と呼ばれる光の **3** 原色の情報を与え，それぞれを **RGB** に各 **8** ビットずつ， 合計 **24** ビット を使うことが多い． **24** ビットなので合計 約 (d) 万通りの色を表現できるわけだが，人間の目で見分けられる色の数はもっと少ないので，一般にはこれで十分とされる．

1 ピクセルを表現するのに **24** ビットを用いると，たとえば幅 **600** ドット高さ **400** ドットの画像ファイルを表現するためには，単純計算で約 (e) **K** バイトのデータ量が必要になるが，画像の質や色調を損なわずにデータ量を小さくする表現形式が利用される．良く用いられる表現形式としては， **GIF** や **JPEG** がある．

第 **6** 問 **(10** 点**)**

以下の文の空白部分 **(a-b)** を埋めなさい.

ホームディレクトリにあるすべてのファイルの属性を表示するには，

```
ls -al ~
```

を実行すれば良い．この結果を **/tmp/cal.1752.9** というファイルに保存するには，シェルのリダイレクション機能を使って

```
ls -al ~ (a)
```

のようにする (カレントディレクトリは **/tmp/** 以外にあるとする)．このコマンドの実行結果の行数を数えるには，リダイレクション機能を使ってファイルを作ってから， **wc** コマンドを使って，

```
wc /tmp/cal.1752.9
```

としてもよいが， **wc** はファイル名を指定しない場合は標準入力を対象とするので，シェルのパイプ機能を使って，

```
ls -al ~ (b)
```

とすることができ，こちらの方がスマートである．

第 **7** 問 **(20** 点**)**

下のように **1** 行ごとに，都道府県名，都道府県庁所在地名，人口 (単位 人)，面積 (単位 km^2) を記録したファイル **Statis** がある (途中を略してある)．

```
Hokkaido  Sapporo     5692321 83452
Aomori    Aomori      1481663  9606
Iwate     Morioka     1419505 15278
(途中略)
Oita      Oita        1231306  6337
Miyazaki  Miyazaki    1175819  7734
Kagoshima Kagoshima   1794224  9186
Okinawa   Naha        1273440  2266
```

コマンド **grep** を使って，以下のような行を抽出する方法を示しなさい (検索パターンだけでなく入力するコマンド行全体を書くこと)．

1. 都道府県庁所在地名の頭文字が **S**
2. **7** 文字以上の都道府県名
3. 都道府県名の頭文字が **S** で，都道府県庁所在地名の頭文字が **N**
4. 面積が **5000** km^2 以上 **10000** km^2 未満

注) grep の検索パターンを以下に簡単にまとめる．

- 任意の 1 文字はピリオドで示す．
- ある文字列に含まれるどれか一つの文字は [文字列] で示す． たとえば ”[aeiou]” は英字の母音文字の一つを示す．
- ある範囲内にある一文字は [文字 - 文字] で示す．たとえば ”[0-9]” は数字一文字で，”[0123456789]” と同じ．
- 「直前にある一文字 (を表すもの) 」の 0 回以上の繰返しを ”*” で表す．たとえば ”ee*” は e, ee, eee, eeee, ... を表し， ” [0-9][0-9]*” は数字の列を表す．
- 「直前にある一文字 (を表すもの) 」の k 回以上 m 回以下の繰返しを ”\{k,m\}” で表す．ちょうど k 回の場合は ”\{k\}” である．
- 行の頭は ”^”，行末は ”$” で表す．