

手抜きのためのプログラム ～英語穴埋めテストCGI～

奈良の山奥で修行中
森若和雄
kazuo-mo@is.aist-nara.ac.jp

あらすじ

- どういう仕掛けか。
- 問題データ形式とそのデータ形式にした理由
- 問題データをパースする正規表現の解説
- テンプレート+置換でhtmlは分離しよう
- 全て動的に実行して、モデルを単純にした
- 全て動的に…って何？ 良い所と悪い所、このシステムにおける評価
- 好評だった点
- いまいちだった点
- せつないオチ

どういう仕掛けか

学生から見ると…

- ログイン→問題→採点→問題→採点→問題→採点→問題→採点→…というドリル形式
- 先生から見ると…
- 問題データを置く→学生がせつせと穴埋めする→CSVファイルに結果が集まる

問題データ形式

\$キーワード={内容}という形式にする。
「内容」の中で{,;}に囲まれた部分は、穴埋めの穴となる。
{,;}以外の文字はそのままHTMLに埋め込まれる。

例:

```
$title={ Apple's iMac arrives}  
$question={update Apple's {iMac} consumer computer  
arrives at stores and resellers but not everybody can get  
one. <p> August 15, 12:45 p.m. P.T in Computing}
```

問題データ形式の決定理由

普通の英文には {,;}はないので、区切り記号は{}にした。
雛型にテキストを埋める為、常に説明つきとなり負担が減るので\$question={}という形式にした。
問題データ作成時に、複数の記号について注意するのは面倒なので穴空けも{,;}にした。
→結果的に、正規表現で簡単に構文解析することもできた。

問題データをパースする

まず\$question=の後の問題本文を取得。他の \$キーワード={内容} も同様

```
$data =~ m^\$question\s*=\s*\{\{\{\{\{\{\{\}\}\}\}\}\}\}\}\s*/is;
```

次に内部を順に q問題番号 という名前のINPUT文に置換

```
while ($question =~ s!\{\{\{\}\}\}\!\<INPUT TYPE\=\TEXT  
NAME\=\"q${i}\">\>!){  
  $i++;  
}
```

問題データをパースする(2)

回答のチェック時には、

```
while ($question =~ m!\{([\^}+)]\}g) {  
  $ans = $1;  
  $reply = $form{"q$1"} || "(none)";  
  のように、$ans に正答、$replyに回答を入れて  
  正答ならO、  
  $result[$1] = "<IMG ALT='O' SRC='img/OK.gif'>[$ans]";  
  誤答ならX  
  $result[$1] = "<FONT COLOR='##ff0000'><IMG ALT='X' SRC='img/NG.gif'>$reply</FONT>[$ans]";  
  に置換します。
```

雛型 + 置換でhtmlはCGIから分離しよう

適当に read_template のような関数を作ってhtmlの雛型を読む

```
$html = read_template("answer.html");
```

あとはひたすら置換

```
$html =~ s/\$question/\$question/ig;  
$html =~ s/\$title/\$title/ig;
```

そして出力

```
print "Content-type: text/html\n\n";  
print $html
```

全て動的に実行して、モデルを単純にした

普通だと、

- 問題データから、フォームと回答とを先生に生成してもらい、採点CGIで採点&記録

でも今回は、

- 問題データから、フォームと回答とを「動的に」生成して、採点CGIで採点&記録

理由は？



運用が楽！

全て動的に。いいところと悪いところ

- いいところ
 - 元データと出力結果の同期を考える必要なし
 - よって問題作成、管理者の手間が減る
- 悪いところ
 - 静的に出力するより1000倍くらい遅い
 - 一度にperlを起動する為にメモリが必要
- 今回は
 - 最大1クラス40人程度返しか使わないことが決まっている
 - 先生はかなりの素人(笑);

反省してみる

- 好評だった点
 - データ簡単
 - 出力がCSVなのでexcelで簡単に集計、分析できる
 - 評価時の OとX

O X

- いまいちだった点
 - 問題毎の回答時間を計測したかったらしいができなかつた(;-;)

せつない才子

- 仕事とってきたのは友人某。
- 彼はサーバにFreeBSDとsambaをインストール
- 私は設計&プログラム作成、改造...



報酬は5.5でした(涙;; くらい)。