

Perlでバイナリファイルを読むサンプルコード

2009年09月02日 13時15分13秒

「読む例は～？」のツッコミがメールで今頃入ったのでどうかと思ったけど記述。

ソース

s2.pl

```
## バイナリファイルを読み込んで16進数の文字列で書き出すサンプル

binmode(STDIN);
$cnt = 0;
$addr = 0;
while(1==1) {
    printf("%08x ", $addr) if (0==$cnt) ;

    last if undef == read(STDIN, $code, 1);
    printf("%02X ", unpack("C", $code) );
    $cnt++;

    if (16==$cnt) {
        print "\n";
        $addr+=$cnt;
        $cnt=0;
    }
}
print "\n--- end\n";
```

標準入出力を使うことを想定した。16進ダンププログラム風。

突っ込み対応。長さゼロのファイルの時はアドレス表示も出ません。

s2a.pl

```
## バイナリファイルを読み込んで16進数の文字列で書き出すサンプル

binmode(STDIN);
$cnt = 0;
$addr = 0;
while(1==1) {
    last if undef == read(STDIN, $code, 1);

    printf("%08x ", $addr) if (0==$cnt) ;
    printf("%02X ", unpack("C", $code) );
    $cnt++;
}
```

```
if (16==$cnt) {
    print "\n";
    $addr+=$cnt;
    $cnt=0;
}
}
print "\n--- end\n";
```

好きなほうをどうぞ。個人的にはアドレスが出てデータが無い行があっても全然構いませんです。

入力データ

input.txt

```
□□□□□
```

いわゆるCP932(SJIS)の文字を5個並べた。

実行結果

```
f:\work>type input.txt
□□□□□
f:\work>perl s2.pl < input.txt
00000000 81 40 81 50 81 60 81 50 81 47
--- end

f:\work>
```

説明

標準入力をバイナリモードにするため binmode()でSTDINを指定した。次に標準入力から1バイトずつ読み、それを unpack()で“ 1文字 ”に変換する。あとは printf()の書式で %02X を指定し16進数2桁で表示。 2バイトのコードの上位 下位バイトを各々長さ1バイトの文字として編集した

旧ページコメント

- ふむふむ。16バイトきっちりのときは、アドレスだけの行が出力されるんですね(^ ^;- JK (2009年08月31日 22時17分13秒)
- 意味がやっとわかった。答えはYES□問題なし。とりあえずその突っ込みような解答は書いておきますね。 - 努力中の人 (2009年09月01日 07時17分32秒)
- 2種類掲載したか□てか、個人的には、ループそのものは無限ループ形式で、ループの中で意図的

に外にジャンプしないとならない作りは□ま、いっか□□ - JK (2009年09月02日 13時15分13秒)

[Perl](#), [技術資料](#), [バイナリファイル](#)

From:

<https://wiki.hgotoh.jp/> - 努力したWiki

Permanent link:

<https://wiki.hgotoh.jp/documents/perl/perl-0002>

Last update: **2024/11/01 16:30**

