

# 情報科学演習 資料 6

## シェルのメタキャラクター

令和元年 5 月 27 日

### 目次

1	コマンド行における特殊文字の扱い	1
1.1	メタキャラクター	1
1.1.1	？と＊	1
1.1.2	その他のメタキャラクター	3
1.2	特殊文字の無効化	3
1.2.1	例	4
1.3	問題	4

## 1 コマンド行における特殊文字の扱い

コマンドを入力する行(コマンド行)の内容を解釈して、コマンドを実行するのはシェルという種類のプログラムでした。ここでは、コマンドに引数を与えて実行する際に必要となる、シェルにとって特別な意味のある文字の取り扱いを学びます。また、検索等でも必要となる、文字列のマッチングに関する考え方の基礎を習得します。

### 1.1 メタキャラクタ

カレントディレクトリに、仮に a1, a2, ..., a5 という 5 つの通常のファイルと distdir というディレクトリのみが存在するとし、a1, ..., a5 を distdir に移動したいとします。ファイル移動のコマンドは mv であり、mv の引数には複数のファイルを指定できますから、

```
mv a1 a2 a3 a4 a5 distdir
```

を実行すれば希望は叶えられますが、移動するファイルをすべて列挙するのは面倒です。

この例のように、一度のコマンド操作で複数のファイルを処理したいとき、メタキャラクタ (*meta-character*) という、シェルにとって特別な意味のある文字を利用すると便利です。メタキャラクタを用いると、上記の操作は

```
mv a? distdir
```

や

```
mv a* distdir
```

で済みます。a? と a\* は、シェルによって共に a1 a2 a3 a4 a5 に展開 (置き換え) されてから、引数として mv コマンドに渡されます。

#### 1.1.1 ? と \*

? と \* は、一言でいえば、

? : 任意の 1 文字

\* : 任意の文字列

にマッチする<sup>1</sup>メタキャラクタです。その具体的な使い方を次の例で見えます。

##### 1. 準備

- (a) カレントディレクトリを /pub/eis/metawork に変更してください。うまくいったかどうか、pwd コマンドで確認してください。
- (b) ls コマンドを使ってカレントディレクトリに存在するファイルを確認してください。次のファイルがあります。

```
A A1 B C a a1 a11 a12 a13 a2 a3 b b1 b11
```

<sup>1</sup>後に説明しますが、? と \* とともに、ファイル名先頭の. (ドット) や / にはマッチしません。

## 2. ? の使い方

echo コマンドは、引数に与えた文字列をそのまま表示するコマンドでした。例えば、

```
echo abc d e
```

を実行すると、画面に abc d e と表示されます。この echo コマンドを使い、? を含む引数がどのようなファイル名に展開されるのかを調べましょう。

(a) 次のコマンドを順番に実行してください。

```
echo ?
echo ??
echo ???
echo ????
```

echo の引数に ? を与えても、? とは表示されません。

シェルは ? をカレントディレクトリに存在する 1 文字からなるファイル名のリストに展開し、引数として echo コマンドに渡します。その結果として 1 文字からなるファイル名がすべて表示されます。

同様に、?? は 2 文字の、??? は 3 文字のファイル名に展開されます。4 文字のファイル名をもつファイルはカレントディレクトリに存在しないので、最後の例では、その旨のメッセージが出ます。

(b) ? を他の文字と組み合わせて使うこともできます。

```
echo a?
echo ?1?
```

最初の例では、ファイル名の先頭が a であって、その後に何という文字でも構わないから、1 文字が続くものに展開されます。その次の例は、1 の前後に任意の 1 文字が存在するファイル名に展開されます。

(c) ? はパス名の一部として使うことができます。例えば、

```
echo /bin/????
```

だと、/bin というディレクトリに存在するファイルのうち、4 文字のファイル名をもつものが該当します。

なお、? はファイル名の先頭にある . や、パス名の区切り記号 / にはマッチしません。

## 3. \* の使い方

\* は ? に似ていますが、1 文字のみではなく、任意の長さの文字列にマッチする点が異なります。

次の例は、各々 a と a1 で始まるファイル名に展開されます。

```
echo a*
echo a1*
```

最初の例で注意すべきは、展開されたファイル名の中に a が含まれていることです。すなわち、任意の長さには、0 文字の長さも含みます。同じ理由から、2 番目の例でも、a1 が結果に含まれます。

次の結果も、容易に予想できるでしょう。

```
echo *3
echo *
echo /bin/*
```

なお、? の場合と同様に、\* はファイル名の先頭にある . や、パス名の区切り記号 / にはマッチしませんので、これらは明示的に記述する必要があります。例えば、カレントディレクトリに存在する . で始まるファイル名をすべて扱うには .\* とする必要があります。

以上、? や \* がどのようなファイル名に展開されるのかを調べるために echo コマンドのみを使ってきましたが、メタキャラクタはシェルが解釈する特殊文字なので、コマンドの引数一般に利用することができます<sup>2</sup>。

例えば、カレントディレクトリに存在する a で始まるファイルを調べたければ、ls a\* を実行するのが自然でしょう。なお、ls の引数にファイル名を与えると、そのファイルの情報のみが表示されます。

### 1.1.2 その他のメタキャラクタ

ファイル名やディレクトリ名に展開される、\* や ? 以外のメタキャラクタとして、次のものもあります。

```
[c1c2...]      [ ] 内の 1 文字 ([c1-c2] を用いて文字範囲を指定することも可能)
{string1,string2,...}  { } 内の文字列
```

これらは次のように使います。

```
echo [abc]
ls [abc]
echo [A-Z]*
echo /{bin,usr,var}
ls /{bin,usr,var}
```

[abc] は a, b, c のいずれかのファイル名に、[A-Z]\* は大文字で始まるファイル名に展開されます。/{bin,usr,var} は /bin /usr /var に展開されます。

また、~ は自分のホームディレクトリの絶対パス名に展開される特殊文字であり、ほとんどのシェルで利用できます。次のように使えます。

```
echo ~
ls ~
echo ~/a*
```

## 1.2 特殊文字の無効化

echo コマンドを使って、画面に What is your name? と表示するにはどうすればいいでしょう。

<sup>2</sup>Windows では、コマンドプロンプトを起動すれば、キーボードからのコマンド入力が可能であり、そこでも \* や ? を利用できます。Windows では \* や ? はワイルドカードと呼ばれます。ただし、コマンドプロンプトでのワイルドカードは、シェルではなく、各コマンドが解釈するものなので、ワイルドカードが利用できるかどうかは実行するコマンドに依存します。

```
echo What is your name?
```

を、結果を予想してから実行してみましょう。

メタキャラクタである `?` を、普通の文字として `echo` コマンドで出力するには、シェルによる `?` の展開を抑止 (エスケープ) してから、`echo` コマンドの引数に与える必要があります。

また、シェルが特殊な意味を持つ記号として解釈する文字には、`$` や `"`、空白等、これまでに紹介した以外にも存在します。コマンド行において、それらを普通の文字として扱う場合にも、次のようにして特殊文字の持つ意味を無効にする必要があります。

1. 単一の特殊文字 `c` に対する特殊な意味の無効化

```
\c
```

2. 文字列 `string` 内の特殊文字に対する特殊な意味の無効化

```
'string'
```

- `string` 内に `'` を含む場合、この方法は使えません。
- 空白を含む文字列を、空白で区切られた複数の引数としてではなく、単一の引数としてコマンドに与えたい場合、この方法は便利です。(引数の区切り文字としての空白文字の意味を無効にします)

### 1.2.1 例

```
echo What is your name\?
echo 'What is your name?'
```

前者の場合、`echo` の引数が 4 個であるのに対し、後者では一つです。

時間に余裕があれば、`tcsh` のオンラインマニュアルを読むことも有意義です。

## 1.3 問題

1. カレントディレクトリを、ホームディレクトリ内の `eis19` に変更した上で、ディレクトリ `/pub/eis/metawork` に存在する `b` で始まる名前のファイルを、一度の操作で全てカレントディレクトリに複製してください。
2. ホームディレクトリにファイルがたくさんある人はホームディレクトリ内を整理してください。ディレクトリを作ってから、メタキャラクタを使ってファイルを移動したり、不要なファイルを削除しましょう。

注意：コマンドの実行結果を取り消すことは基本的にはできませんので、ファイル操作は慎重に行ってください。特にメタキャラクタを使ってファイル操作を行う際には、メタキャラクタの展開結果を事前に確認する習慣を身につけましょう (例えば `ls pp*.c` をまず実行してみるなど)。

3. `echo` コマンドを使って画面に `What's your name?` と表示する方法を考え、いろいろと試してみてください。

4. `cp a b c` と `cp 'a b' c` は、各々、何をするための操作かを考えてください。

実際に操作を試したければ、これらを実際に行うために必要なファイル等を `echo` コマンド等で作成して行ってください。

(ヒント：ファイル名には空白文字を含めることも可能です。ただし、この問を解く以外の場面では推奨しません。)