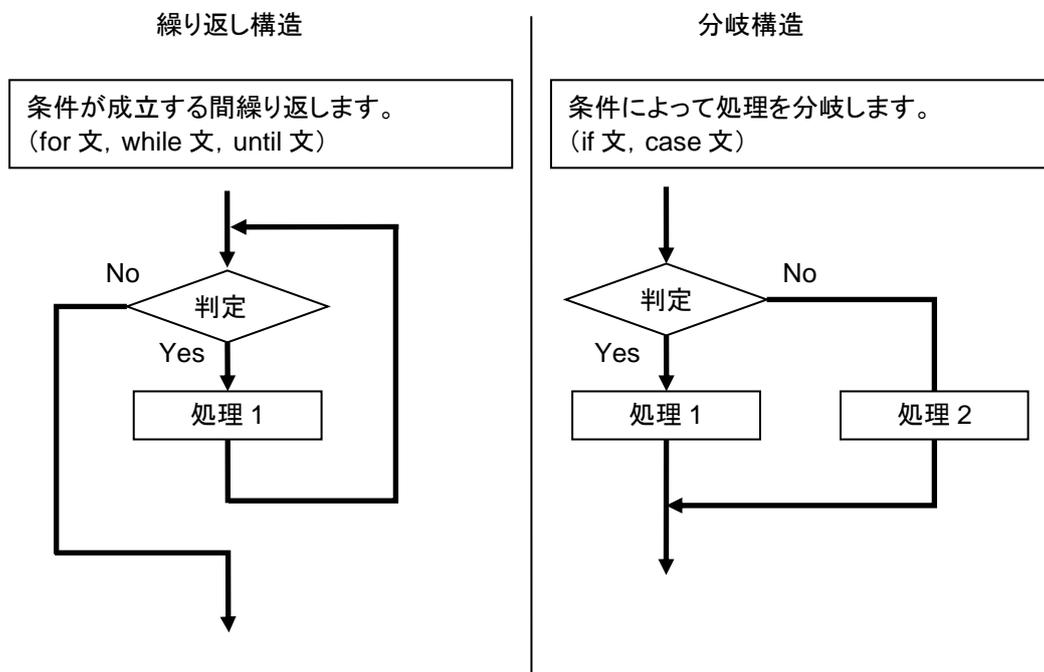


3.4 制御用構文の利用

シェルスクリプト内のコマンドは、記述した順番に実行されますが、「制御用構文」を用いることで、繰り返し同じコマンドの実行や、条件により異なるコマンドの実行ができます。

以下は、主な制御用構文です。

構造	制御用構文
繰り返し	for 文, while 文, until 文
分岐(場合分け)	if 文, case 文



3.4.1 for 文

for 文は、複数の文字列に対して、同じコマンドを繰り返し実行する場合に利用します。

<for 文の書式>

```
for 変数 [ in 文字列1 文字列2 文字列3 ... ]
do
    コマンドリスト
done
```

※[]内は省略可能です。

in に続く文字列を先頭から順に変数へ代入し、do から done までのコマンドを実行します。これを、文字列がなくなるまで繰り返します。

```
$ loop1
a
bb
100
$
```

【loop1 スクリプト】

```
01: #!/bin/sh
02:
03: for i in a bb 100
04: do
05: echo $i
06: done
```

in 以降を省略した場合は、スクリプト実行時の位置パラメータ(\$1~)を先頭から順に変数へ代入し、コマンドの実行を繰り返します。「in \$*」を指定した場合と同様です。

```
$ loop2 a bb 100
a
bb
100
$
```

【loop2 スクリプト】

```
01: #!/bin/sh
02:
03: for i 
04: do
05: echo $i
06: done
```

■ 例題6

引数に指定したファイル名の、ファイル名と内容を順に表示するスクリプト「showfile2」を作成します。ただし、ファイル名は複数指定ができるようにします。

```
$ showfile2 loop1 member
--- loop1 ---
#!/bin/sh

for i in a bb 100
do
    echo $i
done
--- member ---
Line1 : YOUHEI    m  s42-10-06
Line2 : MIKA      w  s36-07-21
Line3 : AYUMI     w  s39-05-07
Line4 : TADAO    m  s41-08-18
Line5 : TOMOKO   w  s40-01-07
Line6 : MITSUYASU m s41-05-30
$
```

【showfile2 スクリプト】

```
01: #!/bin/sh
02:
03: for i
04: do
05:   echo "--- $i ---"      # filename
06:   cat $i                 # print
07: done
```

解説

03-07 : 変数 *i* へ位置パラメータ(\$1、\$2、…)を順番に代入し、do から done までのコマンドを繰り返し実行します。

■ 確認しよう

以下の実行結果が表示されるようにシェルスクリプト「loop1」を変更してください。

```
$ loop1
a
---
bb
---
100
---
NEC
---
$
```

<変更点>

- 「a」、「bb」、「100」の後に「NEC」を表示します。
- 「a」、「bb」、「100」、「NEC」と併せて---(区切り線)も繰り返し表示します。