ksforloop

Nova De Hi

2015년 5월 18일, v0.3

요 약

정수 기반의 for-형 반복문 `\ksforloop`를 정의한다.

차 례

1 사용법

```
\ksforloop{<cnt>=<start>+<step>}{<end>}{<code>}
```

`<cnt>`는 `i`, `j`나 그밖에 원하는 카운터 변수 이름을 쓴다. 이 카운터는 내부적으로 사용되므로 미리 선언할 필요 없다.

`<start>`, `<step>`, `<end>`는 모두 숫자(정수)이다.

for 블럭 안에서 지역적으로(locally) `\ksfor<cnt>` 명령(cs)이 정의된다. 즉, `\ksforloop{i=1+1}{10}{do}`라고 하였을 때 이 반복문 블럭 안에서 `\ksfori`라는 변수를 쓸 수 있다. 현재 진행되는 반복 횟수를 의미한다. 이것은 수(number)가 아니라 콘트롤 시퀀스이므로 정수로서 사용하려면 `\number`를 앞에 붙여준다.

`<code>`는 `\par`를 포함할 수 있다. 그러나 verbatim 텍스트는 올 수 없다.

```
\ksforquit
```

강제로 루프를 종료한다.

`\ksforloop*` 명령은 loop로 실행될 부분을 그루핑하지 않는다. scope가 문제가 될 때 사용할 수 있다.

2 예제

첫 번째 인자를 비우면 `{i=1+1}`한 것과 같다.

```
\ksforloop{}{10}{\ksfori\ }
```

1 2 3 4 5 6 7 8 9 10

루프 카운터를 지정하는 표준적인 사용법은 다음과 같다.

```
\ksforloop{ct=1+1}{10}{\ksforct\ }
```

1 2 3 4 5 6 7 8 9 10

다음 보기에서 `\par`를 쓸 수 있는 것, `\ksforquit`의 사용, `\ksfor<cnt>`를 `\ifnum`으로 체크하는 것을 테스트한다. `\ksforquit`이라는 미리 정의된 명령이 있기 때문에 카운터 이름을 `quit`으로 하면 안 된다.

```
\newcount\Result
\ksforloop{i=1+1}{10}{%
    i = \ksfori,
    \Result = \numexpr\Result+\number\ksfori\relax
    sum = \the\Result
    \ifnum\number\ksfori=5 \ksforquit \fi\par
}
```

i = 1, sum = 1
i = 2, sum = 3
i = 3, sum = 6
i = 4, sum = 10
i = 5, sum = 15

중첩도 허용한다.

```
\ksforloop{i=0+2}{10}{%
    \ksforloop{j=1+1}{5}{%
        \ksfori, \ksforj\ ||
    }\par
}
```

0, 1 || 0, 2 || 0, 3 || 0, 4 || 0, 5 ||
2, 1 || 2, 2 || 2, 3 || 2, 4 || 2, 5 ||
4, 1 || 4, 2 || 4, 3 || 4, 4 || 4, 5 ||
6, 1 || 6, 2 || 6, 3 || 6, 4 || 6, 5 ||
8, 1 || 8, 2 || 8, 3 || 8, 4 || 8, 5 ||
10, 1 || 10, 2 || 10, 3 || 10, 4 || 10, 5 ||

3 팁

tabular의 내용을 채우기 위해 forloop을 써야 할 일이 있다. 다음은 그럴 경우의 한 예이다.

```
\protected\def\tnlinehline{\tabularnewline\hline}
\def\test{ A & B & C \tnlinehline }
\makeatletter
\ksforloop{}{4}{%
  \ksforloop{j=1+1}{3}{%
    \protected@xdef\test{\test
      (\ksfori,\ksforj) \ifnum\number\ksforj=3 \else &\fi
    }}%
  \protected@xdef\test{\test \tnlinehline}%
}
\makeatother
\begin{tabular}{|c|c|c|}
\hline
\test
\end{tabular}
```

A	B	C
(1,1)	(1,2)	(1,3)
(2,1)	(2,2)	(2,3)
(3,1)	(3,2)	(3,3)
(4,1)	(4,2)	(4,3)

4 참고

plain TEX의 반복문 `\loop...\repeat` 외에, LATEX에서 for 스타일 loop문은 다음과 같은 것이 있다.

- forloop 패키지의 `\forloop` 문.
- multido 패키지의 `\multido` 문.
- tikz의 `\foreach` 문.
- xfor의 `\@for` 문.

이 가운데 xfor의 `\@for` 문은 LATEX 커널 명령과 마찬가지로 리스트에 대하여 적용되는 것이므로 논외로 한다. forloop 패키지의 `\forloop`는 예컨대 다음과 같은 방식으로 쓰는데,

```
\newcounter{ct}
\forloop{ct}{1}{\value{ct} < 10}{ \arabic{ct} }
```

새로운 카운터를 직접 만들어야 한다는 점, 그리고 ifthen의 조건문을 주어야 한다는 것이 for-스럽지 못하다.

`\multido`는 매우 훌륭하다. 다양한 설정이 가능하다. 다만 반복문에 `\par`를 직접 쓸 수가 없는데(확장시켜서 하는 방법은 있다) 이따금 불편을 느끼던 차에 이것을 만들게 되었다.

`\multido`만큼 복잡한 것은 아니지만 `\forloop` 대용으로 그럭저럭 쓸 만하다.

5 Known Problems

- tabular, array 환경 안에서 동작하지 않음. 3절을 보라.
- verbatim 텍스트 사용 불가.