Jan Rusinek

ZMIENNE LICZBOWE W TeX-u. ZASTOSOWANIE W DYDAKTYCE

[**Słowa kluczowe.** TeX, deklaracja zmiennych, organizacje egzaminˇw, testy]

Streszczenie.
W pracy przedstawione jest zastosowanie zmiennych liczbowych i działań algebraicznych w TeX-u do tworzenia zadań i testów z różnymi parametrami liczbowymi. Zademonstrowany jest też sposób konstrukcji i używania danych rzeczywistych w postaci dziesiętnej i ułamkowej oraz podobny sposób dla danych tablicowych.

1. Zmienne liczbowe w TeX-u

Możliwości TeX-a wykorzystywania zmiennych liczbowych i manipulowania nimi są teoretycznie skromne. Ale nawet te skromne możliwości pozwalają po stworzeniu odpowiednich procedur uzyskać ciekawe i przydatne rezultaty, z których kilka zareprezentujemy.

Interesujące nas zmienne deklarowane w TeX-u mogą być następujące:

a) zmienne całkowitoliczbowe deklarowane poleceniem `\newcount\zmienna`. Np. deklaracje

```
\newcount\liczba
\liczba=2009
```

rezerwuje w pamięci TeX-a miejsce dla zmiennej liczbowej pod nazwą `\liczba`, a następnie przypisuje jej wartość 2009.

b) długości deklarowane poleceniem `\newdimen\zmienna`. Zmienna ta może być mierzona w punktach (`pt`), centymetrach (`cm`), milimetrach (`mm`), i w tzw. długościach względnych, których tu nie będziemy wykorzystywać. Np. deklaracje

```
\newdimen\nowadlugosc
\nowadlugosc=1.35cm
```

rezerwuje w pamięci komputera miejsce dla zmiennej `\nowadlugosc` i przypisuje jej wartość 1.35cm[103]

2. Operacje na zmiennych

Na powyższych zmiennych możemy przeprowadzać następujące operacje:

a) Dodawanie liczby do liczby. Używamy do tego polecenia

```
\advance\liczba by ....
```

W miejsce `...` można wstawić konkretną liczbę lub zmienną liczbową.

Na przykład kolejne deklaracje

```
\newcount\aa \newcount\bb
\aa=3 \bb=7
\advance\aa by2 \advance\aa by-\bb
```

powodują, że w tym miejscu zmienna `\aa` ma wartość -2.

b) Dodawanie długości do długości. Działa to identycznie jak w przypadku liczb. Na przykład

```
\newdimen\aaa \newdimen\bbb
\aa=3.3cm \bb=22mm
\advance\aaa by2.4cm \advance\aaa by-\bbb
```

spowoduje, że zmienna `\aaa` przyjmie wartość 3.5cm.

c) Mnożenie liczby lub długości przez liczbę całkowitą. Służy do tego polecenie

```
\multiply\zmienna by ....
```

W miejsce `...` można wstawić konkretną liczbę lub zmienną. Działa to identycznie jak w przypadku dodawania.

Można również uzyskać mnożenie długości przez liczbę dziesiętną zapisując liczbę dziesiętną przed długością.

Na przykład po wykonaniu poleceń

```
\newdimen\aaa \newdimen\bbb
\aaa=3.3cm \bbb=2.25\aaa
```

długość `\bbb` przyjmie wartość $2.25 \cdot 3.3$ cm.

d) Dzielenie liczby przez liczbę lub długości przez liczbę. Uzyskujemy poleceniem

```
\divide\zmienna by ....
```

[103] Ponieważ TeX wyświetlając wartości długości używa do oddzielania części całkowitej od dziesiętnej kropki, stosujmy taką konwencję.

W miejsce kropek wstawiamy konkretną liczbę lub zmienną liczbową. Przy dzieleniu liczby przez liczbę odrzucana jest część ułamkowa. Na przykład polecenia

```
\newcount\aa\aa=3
\newdimen\bb \bb=3.4cm
\newcount\cc \cc=2
\divide\aa by\cc \divide\bb by\cc
```

powodują, że zmienna `\aa` przyjmie wartość 1, a zmienna `\bbb` wartość 1.7cm.

We wszystkich działaniach można opuścić napis `by`.

e) Porównywanie z wykorzystaniem instrukcji warunkowych `\ifnum` oraz `\ifdim`. Na przykład wykonanie poleceń

```
\newcount\aa \newdimen\bb
\aa = 10 \bb = 20mm
\ifnum\aa > 11 $aa > 11$ \else $aa \leq 11$, \fi
\ifdim\bb > 2cm $bb > 2cm$ \else $bb \leq 2cm$\fi
```

da napis

$aa \leqslant 11$, $bb > 2$cm.

3. Sposób na uzyskanie zmiennych rzeczywistych i manipulowanie nimi

TeX podaje wartości długości z dokładnością 5 miejsc po przecinku wyświetlając je w punktach. Tzn. jeśli zadeklarujemy

```
\newdimen\aaa\aaa=3.4cm
\the\aaa
```

TeX przeliczy 3.4 cm na punkty i wyświetli wynik 96.73918pt.

Można zatem próbować jako zmienne rzeczywiste wykorzystywać wartości długości. Należy znaleźć sposób na wydruk tego bez napisu **pt**. Można to na przykład zrobić przy pomocy następującego makra:

```
\def\usunpt{\expandafter\USUN\the}
{\catcode`p=12 \catcode`t=12\global\def\USUN#1pt{#1}}
```

Wówczas zapis `\usunpt\aaa` spowoduje „połknięcie” napisu „pt” i wydruk tylko liczby 96.73918.

Wykorzystując makro `\usunpt` możemy manipulować liczbami rzeczywistymi i je mnożyć.

Pokażemy to na przykładzie. Powiedzmy, że chcemy wyprodukować następujące zadanie. TeX ma wylosować dwie liczby rzeczywiste z przedziału $[0; 1]$

przemnożyć je przez siebie, zapytać o wynik, a pod spodem napisać odpowiedź i zrobić to w 100 wersjach.

Wykorzystamy pakiet `random.tex` autorstwa D. Arseneau dołączany do standardowej dystrybucji TeX-a. Wywołanie

`\setrannum{\zmienna}{a}{b}`

powoduje wylosowanie liczby całkowitej z przedziału $[a; b]$ i przypisanie wylosowanej liczby zmiennej `\zmienna`. Wykorzystamy też TeX-ową pętlę

`\loop if \repeat` ([2], [4]).

```
\newcount\pierwsza \newcount\druga
\newdimen\zmj \newdimen\zmd
\newdimen\wynik \newcount\wersje\wersje=0
\loop\ifnum\wersje<100
\setrannum{\pierwsza}{0}{10000}
\setrannum{\druga}{0}{10000}
\advance\wersje1
{\par \bf Zadanie \the\wersje.}
Oblicz z dokładnością 5 miejsc po przecinku
\zmj=\pierwsza pt\divide\zmj by10000
\wynik=\zmj\multiply\wynik\druga
\divide\wynik by10000
\zmd=\druga pt\divide\zmd by10000
$\usunpt\zmj\cdot \usunpt\zmd$.
{\bf Odpowiedź:} $\usunpt\wynik$.
\repeat
```

4. Liczby w postaci ułamków

Zaprezentujemy tu makro pozwalające operować w TeX-u ułamkami i wykonywać na nich operacje algebraiczne. W dalszej części pracy będziemy używać niektórych definicji z LaTeX-a ([5], [7]) m.in. do zapisu ułamków poleceniem `\frac`.

Na początek zademonstrujemy zaadoptowany do TeX-a algorytm Euklidesa, dzięki któremu będziemy skracać ułamki.

Najpierw algorytm pomocniczy wyznaczający liczbę n mod m.

```
\def\n#1mod#2m{\aa=#1\bb=#2
\nmodm=\ba aa\divide\nmodm\bb%
```

```
\multiply\nmodm\bb%
\advance\aa-\nmodm%
\nmodm=\aa}
```

A teraz zasadniczy algorytm Euklidesa:

```
\def\euklides#1#2{\aa=#1%
\bb=#2%
\loop\ifnum\bb>0
\n\aa mod\bb m
\aa=\bb
\bb=\nmodm
\repeat
\nwd=\aa}
```

Po wywołaniu makra `\euklides` z dwoma parametrami liczbowymi ich największy wspólny dzielnik jest przypisany zmiennej `\nwd`.

Możemy zdefiniować algorytm skracający ułamki i algorytm je wypisujący.

```
\def\skroculamek#1/#2{
\euklides{#1}{#2}
\licznik=#1\mianownik=#2
\divide\licznik\nwd
\divide\mianownik\nwd}

\def\piszulamek{\ifnum\mianownik=1
\the\licznik
\else
\frac{\the\licznik}{\the\mianownik}
\fi}
```

Wprowadzimy makra realizujące działania algebraiczne na ułamkach i możemy tworzyć zadania wykorzystujące ułamki.

```
\def\dod#1/#2+#3/#4end{\mianownik=#2\multiply\mianownik#4%
\licznik=#1\multiply\licznik
by#4\cc=#3\multiply\cc#2%
\advance\licznik\cc
\skroculamek{\licznik}{\mianownik}}
```

```
\def\mno#1/#2*#3/#4end{\mianownik=#2\multiply\mianownik#4%
\licznik=#1\multiply\licznik#3%
\skroculamek{\licznik}{\mianownik}}

\def\dzi#1/#2:#3/#4end{\mno#1/#2*#4/#3end}
```

Przypuśćmy, że chcemy w 100 wersjach zrobić zadanie następujące:

Oblicz

$$\frac{\frac{a}{b}+\frac{c}{d}}{\frac{e}{f}+\frac{g}{h}},$$

gdzie a–h są wylosowanymi liczbami z pewnych przedziałów.

Możemy to osiągnąć następująco:

```
\loop\ifnum\xj<100\par \advance\xj1
\setrannum{\xa}{1}{3}\setrannum{\xb}{4}{9}
\setrannum{\xc}{1}{5}\setrannum{\xd}{6}{11}
\setrannum{\xe}{1}{3}\setrannum{\xf}{4}{11}
\setrannum{\xg}{1}{7}\setrannum{\xh}{8}{15}
Oblicz $\displaystyle \frac
{\frac{\the\xa}{\the\xb}+\frac{\the\xc}{\the\xd}}
{\frac{\the\xe}{\the\xf}+\frac{\the\xg}{\the\xh}}.$
\dod\xa/\xb+\xc/\xd end
\ee=\licznik\ff=\mianownik
\dod\xe/\xf+\xg/\xh end
\cc=\licznik
\dd=\mianownik
\dzi\ee/\ff :\cc/\dd end
\ \ Odpowiedź: \piszulamek.
\repeat
```

5. Większa liczba zmiennych i tablice

Zmienne całkowitoliczbowe są w TeX-u wykorzystane do numeracji stron, rozdziałów, równań itp., a długości do parametrów składu, dlatego TeX rezerwuje pamięć zaledwie dla mniej niż 300 takich zmiennych. Nie przewiduje też zmiennych w postaci tablic.

Ten problem można jednak rozwiązać. Jednym ze sposobów jest trzymanie informacji o wartości zmiennych w zewnętrznym pliku. W wypadku, gdy zmienne te nie są zbyt często przetwarzane i zmieniane ich wartości, ta metoda jest skuteczna i dobra. Została np. zastosowana przez autora w programie do produkcji testów opisanym w [8] (program do przetwarzania wyników egzaminu jest opisany w [9]), gdzie w zewnętrznym pliku przechowuje się informacje o sposobie permutacji pytań i odpowiedzi w kolejnych zestawach i parametry liczbowe wylosowane do różnych zadań. Tych parametrów jest 9 razy liczba zadań razy liczba zestawów, zatem już przy kilku zadaniach w zestawie i kilku zestawach dopuszczalna liczba zmiennych przechowywanych w pamięci TeX-a zostałaby przekroczona.

Natomiast TeX posiada możliwości trzymania w pamięci 25000 makr (definicji). I je możemy wykorzystać jako zmienne liczbowe lub tablice.

Nazwy zmiennych definiowanych przez `\def\nazwa` w słowie `nazwa` mogą zawierać tylko duże lub małe litery od `a` do `z`. Jest natomiast sposób, który umożliwia tworzenie makr wykorzystujących inne znaki, w szczególności cyfry. Jest to polecenie `\csname ... \endcsname`([4]).

I tak jeśli utworzymy definicję (makro)

`\expandafter\def\csname22\endcsname{123}`,

to TeX dzięki poleceniu `\expandafter`[104] najpierw rozwinie `\csname` do makra `\22`, a następnie przypisze mu ciąg znaków 123, ale nie liczbę 123!

Możemy jednak przekształcić to w liczbę w następujący sposób

`\cc=\csname 22\endcsname`

Od tej chwili licznik `\cc` ma wartość 123 i możemy już wykonywać na nim operacje arytmetyczne.

Ta metoda doskonale się nadaje do zmiennych tablicowych. Oto ilustracja.

Przypuśćmy, że mamy wylosować 1000 liczb z przedziału $[1; 10000]$, umieścić je w 1000 elementowej tablicy, a następnie usunąć wszystkie podzielne przez 3 przesuwając odpowiednio pozostałe wartości w tablicy i na samym końcu zapisać je w postaci $a[i] =$ `odpowiednia wartość`. Oto algorytm (użyjemy pakietu `ifthen.sty` i jego pętli `\whiledo` pozwalającej uniknąć ewentualnych kłopotów związanych z dodatkowymi instrukcjami warunkowymi w pętli `\loop`).

```
\cc=0\nn=1000
\whiledo{\cc<\nn}{\advance\cc1
```

[104] Bardzo dobre opisy poleceń `\csname` oraz `\expandafter` z ciekawymi praktycznymi przykładami można znaleźć w [1] i [3].

```
\setrannum{\xx}{1}{10000}
\expandafter\edef\csname
a\the\cc\endcsname{\the\xx}
} % utworzenie początkowej tablicy
\cc=0
\whiledo{\cc<\nn}{%
\advance\cc1%
\hh=\csname a\the\cc\endcsname%
\n\hh mod3m%
\ifnum\nmodm=0%
\dd=\cc%
\whiledo{\dd<\nn}{\yy=\dd\advance\dd1%
\ww=\csname a\the\dd\endcsname%
\expandafter\edef\csname
a\the\yy\endcsname{\the\ww}%
}
\advance\cc-1\advance\nn-1
\else\fi
}% usunięcie liczb podzielnych przez 3
\cc=0
\whiledo{\cc<\nn}{\advance\cc1%
\ww=\csname a\the\cc\endcsname%
$a[\the\cc]=\the\ww$, }% wypisanie końcowej tablicy
```

6. Parametry liczbowe w pytaniach testowych

W przedstawionej w [8] metodzie tworzenia testów egzaminacyjnych pytania i odpowiedzi mogą być przepermutowane tak, aby każda zdająca osoba miała inny wzorzec odpowiedzi. Składnia pytania testowego jest następująca:

```
\question
Pytanie
\answers
{Odpowiedź pierwsza \trueX}
{Odpowiedź druga \trueX}
...
{Odpowiedź n-ta \trueX}
\endquestion,
```

gdzie za `X` wpisujemy 1, gdy odpowiedź jest prawdziwa, oraz 0, gdy jest fałszywa.

W wypadku zadań dotyczacych przedmiotów „ilościowych” możemy tak formułować treść zadania, aby w każdym zestawie były inne dane liczbowe. W programie w każdym zadaniu przewidziano 9 całkowitoliczbowych zmiennych wylosowanych z zadeklarowanego przedziału. Kryją się one pod licznikami `\xa`, `\xb`, `\xc`, `\xd`, `\xe`, `\xf`, `\xg`, `\xh`, `\xj`. Polecenie `\xi` jest w TeX-u już wykorzystane do greckiej litery ξ. Oto przykład zadania wykorzystującego taką możliwość.

```
\question
Niech $f(x)=\frac{\the\xa x - 1}{\the\xb x + 1}$. Wtedy
$f'(0)$ równa się:
\answers
{\xh=\xa\advance\xh\xb$\the\xh$\true1}
%
{\xh=\xa\advance\xh\xb\advance\xh-2$\the\xh$\true0}
%
{\xh=\xa\advance\xh\xb\advance\xh1$\the\xh$\true0}
%
{\xh=\xa\advance\xh\xb\advance\xh-1$\the\xh$\true0}
\endquestion
```

Treść powyższego zadania jest następująca: trzeba obliczyć pochodną funkcji

$$f(x) = \frac{ax - 1}{bx + 1}$$

w zerze. Parametry a i b kryją się pod poleceniami `\xa` oraz `\xb`. Wynikiem jest

$$f'(x) = \frac{a(bx + 1) - (ax - 1)b}{(bx + 1)^2}.$$

Wstawiając $x = 0$ otrzymamy $f'(0) = a + b$. W pierwszej odpowiedzi mamy prawidłowy wynik, w pozostałych prawidłowy wynik powiększony lub pomniejszony o odpowiednią liczbę różną od zera.

Można też zastosować inny, bardziej pomysłowy algorytm wykorzystania losowo wybranych zmiennych. Mianowicie można `X` po poleceniu `\true` potraktować jak zmienną liczbową i wykorzystując to, nadawać jej wartość w zależności od obliczeń wykonywanych w kolejnych odpowiedziach.

Oto przykład takiego zadania testowego ze statystki matematycznej sprawdzającego umiejętność zastosowania testu χ^2.

```
%1. Test chi kwadrat sprawdza czy czy rozkład jest trzypunktowy
\question
W pewnym kasynie można postawić na trzy kolory: biały, czerwony i
niebieski. Aby sprawdzić ruletkę, która w założeniu ma losować
każdy kolor z takim samym prawdopodobieństwem, puszczono ją w
ruch $30$ razy i otrzymano wyniki dane poniżej:
\xc=30\advance\xc-\xa\advance\xc-\xb
$$\begin{array}{|l|c|c|c|}
\hline
{\rm kolor}&{\rm biały}&{\rm czerwony}&{\rm niebieski}\\\hline
{\rm liczba zatrzymań}&\the\xa&\the\xb&\the\xd\\\hline
\end{array}
$$
\xg=\10\advance\xg-\xa
\multiply\xg\xg
\xe=10\advance\xe-\xb\multiply\xe\xe
\advance\xg\xe
\xe=10\advance\xe-\xd\multiply\xe\xe
\advance\xg\xe
% \xg = 10*wartość statystyki testowej
\answers
%
{Na poziomie istotności $\alpha = 0.005$ odrzucamy
hipotezę {\it ruletka jest uczciwa}
\ifnum\xg>105\xf=1\else\xf=0\fi\true\xf}
%
{Na poziomie istotności $\alpha=0.01$ odrzucamy
hipotezę {\it ruletka jest uczciwa}
\ifnum\xg>92\xf=1\else\xf=0\fi\true\xf}
%
{Na poziomie istotności $\alpha=0.025$ odrzucamy
hipotezę {\it ruletka jest uczciwa}
\ifnum\xg>73\xf=1\else\xf=0\fi\true\xf}
%
{Na poziomie istotności $\alpha=0.05$ odrzucamy
hipotezę {\it ruletka jest uczciwa}
\ifnum\xg>59\xf=1\else\xf=0\fi\true\xf}
\endquestion
%
```

Ruletka została zakręcona 30 razy. Komputer losuje dwie zmienne całkowite: a (kryjąca się pod zmienną TeXową `\xa` – będzie to liczba informująca ile razy wypadł kolor biały, b (zmienna TeXowa `\xb`) (kolor czerwony) i przypisuje zmiennej c (`\xc`) wartość $30 - a - b$ (kolor niebieski).

Przypomnijmy ([6]) zasadę testu χ^2. Obliczamy wartość statystyki testowej:

$$\chi^2_{\text{obl}} = \sum \frac{\text{wartość zaobserwowana - wartość spodziewana}}{\text{wartość spodziewana}}.$$

Odrzucamy hipotezę na poziomie istotności α, jeśli nierówność

$$\chi^2_{\text{obl}} > \chi^2(1-\alpha, n-1)$$

jest prawdziwa.

W naszym przypadku $n = 3$, wartość spodziewana wynosi we wszystkich wypadkach 10, zatem wartość statystyki testowej jest równa

$$\chi^2_{\text{obl}} = \frac{1}{10}\left[(a-10)^2 + (b-10)^2 + (c-10)^2\right].$$

W zależności od wyboru α znajdujemy w tablicach statystycznych wartość $\chi^2(1-\alpha, 2)$: 10.59 dla $\alpha = 0.995$, 9.21 dla $\alpha = 0.99$, 7.38 dla $\alpha = 0.975$, 5.99 dla $\alpha = 0.95$.

Pod licznikiem `\xg` mamy wyliczoną wartość $10\cdot\chi^2_{\text{obl}}$. Jest to zawsze liczba całkowita. Wystarczy zatem ją porównać odpowiednio z liczbami 105, 92, 73, 59 i w zależności od rezultatu przypisać zmiennej `\xf` wartość 1 lub 0 i po poleceniu `\true` wpisać `\xf`.

7. Podsumowanie

Ktoś może zadać pytanie. Po co tworzyć te dodatkowe polecenia wymagające nakładu pracy i pomysłowości, skoro w innych językach programowania to wszystko jest „prawie gotowe"? Odpowiedź jest prosta. Jeśli chcemy tworzyć teksty zawierające wzory matematyczne, żaden inny język ani edytor tekstu nie zapewni właściwej graficznie jakości tych wzorów. TeX został stworzony właśnie z myślą o składzie matematycznym i robi to do czego został przeznaczony w sposób niedościgniony. Zdaniem autora wysiłek włożony w otrzymanie podobnych efektów do prezentowanych w tej pracy może procentować przez wiele lat. Można cały czas modyfikować gotowe materiały czy tworzyć nowe wykorzystując już utworzone makra i algorytmy.

Bibliografia

[1] Abrahams P., Hargreaves K., Berry K., (1990); *TeX for the Impatient*, Addison-Wesley Publishing Company

[2] Borde A., *TeX w przykładach*, `ftp://ftp.gust.org.pl//pub//GUST/Ćontrib//TBE//tbe.pdf`

[3] Eijkhout V., (1992); *TeX by Topics. A TeXnician's Reference*, Addison-Wesley Publishing Company

[4] Knuth D. E., (2005); *TeX Przewodnik użytkownika*, WNT

[5] Lamport L., (1992); *LaTeX: System przygotowywania dokumentów. Przewodnik użytkownika i podręcznik*, Ariel, Kraków

[6] Niemiro W., (1999); *Rachunek prawdopodobieństwa i statystyka matematyczna*, Szkoła Nauk Ścisłych

[7] Oetiker T, Partl H., Hyna I., Schlegl E., *Nie za krótkie wprowadzenie do systemu LaTeX* 2_ε, `ftp://tug.ctan.org//pub//tex-archive//info/łshort//polish`

[8] Rusinek J., (2007); *Algorytm permutowania w TeX-u zastosowany do informatyzacji procesu egzaminacyjnego*, „Rocznik Naukowy Wydziału Zarządzania w Ciechanowie", Tom I, (153-174)

[9] Rusinek J., (2008); *Pliki do odczytu i zapisu w TeX-U – zastosowanie do przetwarzania wyników egzaminu*, „Rocznik Naukowy Wydziału Zarządzania w Ciechanowie", Tom II, (107-124)