

ДАЛЬНЕВОСТОЧНЫЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ НАУК
КАФЕДРА МАТЕМАТИЧЕСКИХ МЕТОДОВ В ЭКОНОМИКЕ

А. С. ВЕЛИЧКО

ПОДГОТОВКА УЧЕБНЫХ ЗАДАНИЙ В ПАКЕТЕ FR

Учебно-методическое пособие

Владивосток
Издательство Дальневосточного университета
2007

УДК 004.912
ББК 32.97
В27

Величко А. С.

В27 Подготовка учебных заданий в пакете FR. Учебно-методическое пособие / А. С. Величко. – Владивосток : Изд-во Дальневост. ун-та, 2007. – 20 с.

В пособии содержатся сведения об использовании пакета fr, входящего в состав издательской системы TeX, для автоматизации создания набора учебных заданий со случайными входными данными. Приводится пример практического использования пакета.

Адресовано преподавателям, сотрудникам учебно-методических подразделений вузов и всем тем, кто в своей деятельности сталкивается с необходимостью создания набора тестовых и контрольных заданий для проверки знаний учащихся.

В $\frac{1504000000}{180(03) - 2007}$

УДК 004.912
ББК 32.97

© Величко А.С., 2007
© ИМКН ДВГУ, 2007

Предисловие

Выполнение контрольных работ, заданий в рамках самостоятельной работы и тестирование по учебным предметам являются необходимыми формами обучения и проверки текущих и остаточных знаний студентов, школьников и всех тех, кто учится.

Выполнение самостоятельной работы в часы самостоятельного изучения также предполагает независимый между студентами одной группы набор учебных заданий. Удачной моделью проведения контрольной работы является предоставление индивидуальных вариантов контрольной работы с независимыми учебными заданиями. При выполнении независимых учебных заданий учащимися возрастает самостоятельность при их выполнении, что сказывается на эффективности освоения учебного предмета, с другой стороны, повышает адекватность полученных учащимися оценок, при этом возникает дополнительный эффект «погони за лидером» в рамках учебной группы.

Особенность учебных курсов, связанных с математическими вычислениями, состоит в высокой потребности *самостоятельного* выполнения учащимися *учебных задач*. Их решение требует разбиения на несколько взаимосвязанных подзадач различных типов и проведения разнородных вычислений. В таком случае важное значение приобретает создание набора учебных задач и заданий.

Зачастую большим препятствием для разработки и подготовки такого набора заданий является высокая трудоемкость этого процесса. Во-первых, она связана с собственно подготовкой в электронном или печатном виде большого числа учебных заданий. Если по рабочей учебной программе ВУЗа предусматривается 4 контрольные работы в учебный семестр с 5 учебными заданиями в каждой для группы студентов из 30 человек, это влечет за собой создание 600 независимых заданий.

Однако, количество заданий - не единственный фактор, усложняющий процесс подготовки учебных заданий. Экономико-математические задачи требуют наличия этапа разработки при подготовке учебных заданий, когда входные данные для задачи ограничены экономическим смыслом используемых переменных (выпуск и цены не могут быть отрицательными величинами) или используемые переменные задачи должны быть связаны каким-то ограничением (валовой национальный продукт, рассчитанный как сумма расходов экономических агентов, должен совпадать с суммой

компонент доходов в экономике) или должны находиться в определенном отношении (например, значение одной переменной обязательно больше значения другой). Еще более сложным является создание таких входных данных, чтобы результаты решения задачи также бы были допустимыми, то есть удовлетворяли каким-то ограничениям вышеперечисленных типов.

Нагрузка на преподавателя возрастает не только при разработке и подготовке, но и при проверке каждой контрольной работы. Изменение программ обучения и взлом банков заданий для компьютерного тестирования приводит к необходимости ежегодного пересмотра набора учебных заданий.

Этап разработки можно упростить, подготовив некоторое количество типовых задач и генерируя большое количество независимых наборов исходных данных. На этапе подготовки набора учебных заданий как раз и может помочь автоматизация.

В данном пособии рассматривается случай подготовки нескольких наборов исходных данных для нескольких типовых учебных задач в автоматическом режиме с использованием свободно-распространяемого программного обеспечения. Рассматриваемый в данном пособии подход к автоматизации, конечно, не решает в полной мере всех задач подготовки независимых учебных заданий, поскольку набор типовых задач все же должен разрабатывать сам преподаватель.

В данном пособии рассматриваются возможности пакета `fr`, используемого в составе издательской системы `TeX`. Данный пакет использовался автором для автоматической подготовки контрольных работ по экономической теории для студентов первого курса специальности «Математические методы в экономике». В пособии приводится пример использования пакета.

Базовые функции пакета `fp`

Пакет `fp` (аббревиатура от англ. *fixed point*) [1] разработан М. Мехличем (*Michael Mehlich*). Данный пакет входит в широко известные свободно-распространяемые дистрибутивы *MikTeX* [2] и *TeX* [3] издательской системы *TeX*. Пакет `fp` распространяется на основе лицензионного соглашения *LaTeX Project Public License* [4].

```
fp-package
```

```
(c) Michael Mehlich 1994 -- 1999
```

```
contributions by: Denis Girou
```

```
acknowledgment to: Denis Girou, Miroslav Balda,  
Ricardo Sanchez Carmenes
```

```
% Copyright 1994 -- 1999 Michael Mehlich
```

```
% This package can be redistributed and/or  
modified under the terms of the LaTeX Project  
Public License Distributed from CTAN archives at  
http://www.latex-project.org/lppl.txt either  
version 1 of the License, or any later version.
```

```
(current) e-mail: mmehlich@semdesigns.com;
```

```
michael@mehlich.com
```

Чтобы использовать пакет `fp` в документе спецификации *latex2e*, необходимо в преамбуле документа вызвать следующую команду:

```
\usepackage[<options>]{fp}.
```

В качестве необязательного параметра `<options>` можно использовать следующие:

- `nomessages` – в этом случае в `.log` файл и на экран при обработке документа не будут печататься сообщения о вызываемых функциях,
- `debug` – при указании данной опции будут выводиться дополнительные отладочные сообщения.

Команды пакета указываются в теле документа между тэгами `/begin{document}` и `/end{document}`.

Запретить или разрешить вывод сообщений и отладочной информации можно следующими специальными командами:

- `\FPmessagestrue` - разрешает вывод сообщений,
- `\Fpmessagesfalse` - запрещает вывод сообщений,

- `\FPdebugtrue` - разрешает вывод отладочной информации,
- `\FPdebugfalse` - запрещает вывод отладочной информации.

Пакет `fp` работает с числами фиксированной точности (18 знаков до и столько же после запятой) в допустимом диапазоне от `-99999999999999999999.99999999999999999999` до `99999999999999999999.99999999999999999999`.

Переменные в `fp` указываются двумя способами, когда специально не оговорено иное. Во-первых, обозначение переменной может начинаться с символа `\` (backslash), за которым указывается имя переменной. Такое имя должно удовлетворять стандартным требованиям к строковым константам и не содержать цифр на конце. Например, переменные `\x` и `\omega`. Во-вторых, имя переменной может указываться в фигурных скобках, например `{omega}`, в частности, когда оно содержит цифры на конце.

Выражения, обозначающие операции над переменными и числами или просто числа заключаются в фигурные скобки, например `{-3}` и `{2*z+3}`.

Все команды пакета `fp` начинаются с символа `\` (backslash) и заглавных букв `FP` без пробелов, т.е. `\FP`. Рассмотрим базовые команды пакета.

Команда `\FPset` осуществляет присваивание переменной какого-то числа или другой переменной. Например, команда `\FPset\x{1}` выполнит присваивание $x=1$, а `\FPset\x{y}` присвоит значение, хранящееся в переменной `y`, переменной `x`.

Команда `\FPprint` предназначена для вывода значения переменной или числа (но не выражения!). Например, `\FPprint\z` (или `\FPprint{z}`) и `\FPprint{2.3}` выведут соответственно числовое значение переменной `z` и число `2.3`. Заметьте, что значение переменной `z` и число `2.3` будут выведены с 18 цифрами после запятой (арифметика с фиксированной точностью!), поэтому необходимо использовать функции `round`, `trunc` или `clip`, описываемые далее, для указания требуемой точности вывода числа.

Операции сложения, деления, умножения и вычитания соответственно `\FPadd`, `\FPdiv`, `\FPmul`, `\Fpsub` имеют три аргумента. Первый определяет имя переменной, в которой будет записан результат операции, а два последующих – операнды, которые

могут быть только числами, указываемыми в фигурных скобках, или переменными, начинающимися с символа \ (backslash).

Команды пакета `fp`, начинающиеся с `\FPif`, соответствуют условному оператору языков программирования и могут содержать один или два операнда. Например, команда

`\FPiflt x y \FPprint{a} \else \FPprint{b} \fi`

соответствует выполнению `\FPprint{a}`, если $x < y$, и `\FPprint{b}` при невыполнении условия $x < y$.

В таблице приводятся условные операторы пакета и типы проверяемых ими условий.

Таблица 1. Условные операторы в `fp`.

Команда пакета <code>fp</code>	Тип условия
Бинарные операторы	
<code>\FPiflt</code>	$x < y$
<code>\FPifeq</code>	$x = y$
<code>\Fpifgt</code>	$x > y$
Унарные операторы	
<code>\FPifneg</code>	$x < 0$
<code>\FPifpos</code>	$x >= 0$
<code>\FPifzero</code>	$x = 0$
<code>\FPifint</code>	x - целое

Команда `\ifFPtest` не содержит операндов и выполняет последний из ранее вызванных условных операторов с теми же операндами.

Уравнения, константы, функции

Следующие команды пакета `fp` предназначены для решения линейного уравнения, нахождения вещественных корней полиномов второй, третьей и четвертой степени. Если некоторые корни отсутствуют (поскольку отсутствует проверка равенства нулю коэффициента при старшей степени) или являются комплексными, выводится предупреждающее сообщение.

`\FPlsolve x a b` присваивает переменной x решение уравнения $ax + b = 0$.

`\FPqsolve x y a b c` присваивает переменным x и y решения квадратного уравнения $ax^2 + bx + c = 0$.

`\FPcsolve\ x\ y\ z\ a\ b\ c\ d` присваивает переменным x , y и z решения кубического уравнения $ax^3+bx^2+cx+d=0$.

`\FPqqsolve\ x\ y\ z\ v\ a\ b\ c\ d\ h` присваивает переменным x , y , z и v решения уравнения четвертой степени $ax^4+bx^3+cx^2+dx+h=0$.

Команда `\FPpi` возвращает число $\pi=3.141592653589793238$, а команда `\FPe` – неперово число 2.718281828459045235.

Рассмотрим команды, которые возвращают значения различных математических функций. Здесь первый операнд является именем переменной, последующие могут быть как переменной, начинающейся с символа `\` (backslash), так и вещественным числом, указываемым в фигурных скобках.

Таблица 2. Математические функции пакета `fp`.

Команда пакета <code>fp</code>	Функция
<code>\FPsin\ z\ x</code>	$z=\sin(x)$
<code>\FPcos\ z\ x</code>	$z=\cos(x)$
<code>\FPsincos\ z\ y\ x</code>	$z=\sin(x), y=\cos(x)$
<code>\FPtan\ z\ x</code>	$z=\operatorname{tg}(x)$
<code>\FPcot\ z\ x</code>	$z=\operatorname{ctg}(x)$
<code>\FPtancot\ z\ y\ x</code>	$z=\operatorname{tg}(x), y=\operatorname{ctg}(x)$
<code>\FParcsin\ z\ x</code>	$z=\arcsin(x)$
<code>\FParccos\ z\ x</code>	$z=\arccos(x)$
<code>\FParcsincos\ z\ y\ x</code>	$z=\arcsin(x), y=\arccos(x)$
<code>\FParctan\ z\ x</code>	$z=\operatorname{arctg}(x)$
<code>\FParccot\ z\ x</code>	$z=\operatorname{arcctg}(x)$
<code>\FParctancot\ z\ y\ x</code>	$z=\operatorname{arctg}(x), y=\operatorname{arcctg}(x)$
<code>\FPexp\ z\ x</code>	$z=\exp(x)$
<code>\FPln\ z\ x</code>	$z=\ln(x)$
<code>\FPpow\ z\ x\ y</code>	$z=x^y$
<code>\FProot\ z\ x\ y</code>	$z=\sqrt[y]{x}$
<code>\FPabs\ z\ x</code>	$z= x $
<code>\FPneg\ z\ x</code>	$z=-x$
<code>\FPsgn\ z\ x</code>	$z=\operatorname{sgn}(x)$
<code>\FPmin\ z\ x\ y</code>	$z=\min(x,y)$
<code>\FPmax\ z\ x\ y</code>	$z=\max(x,y)$

Команда `\FPseed=1` устанавливает счетчик генератора псевдослучайных чисел. Вместо числа 1 можно указать любое другое

вещественное число, в этом случае можно генерировать другую псевдослучайную последовательность. Команда `\FPrandom\ x` записывает в переменную x реализацию равномерно распределенной случайной величины на отрезке $[0,1]$.

Команда `\FPpascal\ x\ {3}` записывает в переменную x третью строку треугольника Паскаля, которую можно вывести на экран командой `\FPprint`. Вместо числа 3 в фигурных скобках можно указать имя переменной, например `\a`, в которой хранится натуральное число от 0 до 63.

Следующие команды позволяют изменить представление выводимого на печать числа. Команда `\FPround\ x\ y\ {3}` присваивает переменной x значение переменной y , округленное до трех знаков после запятой. Команда `\FPtrunc\ x\ y\ {3}` присваивает переменной x значение переменной y , урезанное до трех знаков после запятой путем. Вместо третьего аргумента данных команд, то есть `{3}`, можно указать имя переменной, например `\a`, в которой хранится натуральное число.

Если в процессе вычислений переменная x равна 5, то вывод значения этой переменной путем вызова команды `\FPprint\ x` приведет к печати числа `5.00000000000000000000` (с восемнадцатью нулями после запятой, вследствие использования чисел фиксированной точности). Для устранения этого эффекта используется команда `\FPclip\ x\ y`, которая присваивает переменной x значение переменной y без незначащих нулей в конце записи числа.

Вычисление выражений

В пакете `fp` реализованы функции, позволяющие пользователю работать с выражениями в инфиксной и постфиксной записи.

Команда `\FPupn` предназначена для вычисления значения выражения, записанного в постфиксной нотации (обратной польской записи). Первым аргументом является имя переменной, например `\x`, а вторым – запись вычисляемого выражения, указываемого в фигурных скобках.

Допускается использование знаков арифметических операций `+`, `-`, `*`, `/`, всех математических функций пакета, в именах функций не нужно использовать символы `FP`. Можно использовать следующие специальные команды:

- `pop` - убирает верхний элемент стека,
- `swap` - обменивает содержимое двух первых элементов стека,
- `copy` - копирует значение верхнего элемента стека.

Например, вызов

```
\FPupn\z{17 2.5 + 17.5 - 2 1 + * 2 swap /}
```

присваивает переменной `z` значение выражения

$$((17.5 - (17 + 2.5)) * (2 + 1)) / 2.$$

При вычислении выражений можно использовать переменные, в этом случае имя переменной указывается без символа `\` (backslash), например

```
\FPupn{result}{r - 1 * 0.2 + sin}.
```

При вычислении выражений можно использовать функции `round`, `trunc` и `clip`, которые идентичны по действию вышеописанным командам `\FPround`, `\FPtrunc` и `\Fpclip`. Например,

- `clip(2*x)` означает отбрасывание все незначащих нулей результата вычисления выражения $2x$,
- `round(2*x):3` означает округление результата вычисления выражения $2x$ до трех знаков после запятой,
- `trunc(2*x):3` означает урезание результата вычисления выражения $2x$ до трех знаков после запятой.

Для вычисления значения выражения в инфиксной записи используется команда `\FPeval`. Первым аргументом является имя переменной, например `\x`, а вторым – запись вычисляемого выражения, указываемого в фигурных скобках.

Допускается использование знаков арифметических операций `+`, `-`, `*`, `/` и операции возведения в степень `^`, всех математических функций пакета, в именах функций не нужно использовать символы `FP`. Например, команда для вычисления выражения $\sqrt{\sin 2.5}$

```
\FPeval\result{round(root(2, sin(2.5)):2)}
```

или, что приводит к аналогичному результату, команда

```
\FPeval{result}{round(root(2, sin(2.5)):2)}
```

присвоит переменной `result` значение 0.77.

В результате выполнения команды

```
\Fpeval\result{clip(2*3+5*6)}
```

переменной `result` будет присвоено значение 36 (без незначащих нулей).

Использование пакета `fp` для генерации заданий

Ниже рассматривается практический опыт использования пакета `fp` для автоматической подготовки заданий для контрольных работ по экономической теории со случайными входными данными.

Текст файла `test.tex` контрольной работы

```
\documentclass[12pt,a4paper]{article}
\usepackage[cp1251]{inputenc}
\usepackage[russian]{babel}
\usepackage{fp}
\begin{document}
\input{seed}
\begin{center} Контрольная работа \No 1 \\
Вариант \FPprint{n} \end{center}
\FPrandom\x \FPrandom\y \FPrandom\z
\FPrandom\p \FPrandom\q \FPrandom\r
\FPrandom\xx \FPrandom\yy \FPrandom\zz
\FPrandom\pp \FPrandom\qq \FPrandom\rr
\FPeval\x{round((1+10*x):0)}
\FPeval\y{round((1+10*y):0)}
\FPeval\z{round((1+10*z):0)}
\FPeval\p{round((1+10*p):0)}
\FPeval\q{round((1+10*q):0)}
\FPeval\r{round((1+10*r):0)}
\FPeval\xx{round((1+10*xx):0)}
\FPeval\yy{round((1+10*yy):0)}
\FPeval\zz{round((1+10*zz):0)}
\FPeval\pp{round((1+10*pp):0)}
\FPeval\qq{round((1+10*qq):0)}
\FPeval\rr{round((1+10*rr):0)}
\FPeval\pp{clip(pp+p)} \FPeval\qq{clip(qq+q)}
\FPeval\rr{clip(rr+r)}
\FPeval\xx{clip(xx+x)} \FPeval\yy{clip(yy+y)}
\FPeval\zz{clip(zz+z)}
{\bf Задание 3}. В экономике производится только
три товара: апельсины, автомобили и компьютеры. По
данным, приведенным в таблице, рассчитайте
реальный и номинальный ВВП в 1996 г., считая 1995
```

год базовым, дефлятор ВВП, инфляцию и темп роста реального ВВП в 1996 г. по отношению к 1995 г.

```
\begin{table}[!h]
\begin{center}
\begin{tabular}{|l|c|c|c|c|}
\hline
Годы & \multicolumn{2}{c|}{1995г.} & & \\
\multicolumn{2}{c|}{1996г.} & \multicolumn{3}{c|}{} \\
\hline
Продукты & Цена & Кол-во & Цена & Кол-во \\
\hline
Апельсины & \FPprint{p} & \FPprint{x} & \FPprint{pp} & \FPprint{xx} \\
\hline
Автомобили & \FPprint{q} & \FPprint{y} & \FPprint{qq} & \FPprint{yy} \\
\hline
Компьютеры & \FPprint{r} & \FPprint{z} & \FPprint{rr} & \FPprint{zz} \\
\hline
\end{tabular}
\end{center}
\end{table}
\end{document}
```

Промежуточный файл *seed.tex*, содержит две команды пакета fp:
`\FPset{n}{1}` `\FPseed=1`

Переменная *n*, присваиваемая командой `\FPset`, отвечает за номер варианта контрольной работы (в данном случае 1), а команда `\FPseed` инициализирует генератор псевдослучайных чисел значением 1 (это число должно быть различным для разных вариантов контрольных работ, иначе будут созданы одинаковые наборы входных данных!).

Следующий файл *key.tex* содержит текст ответов к контрольному заданию.

Текст файла key.tex ответов к контрольной работе

```
\documentclass[12pt,a4paper]{article}
\usepackage[cp1251]{inputenc}
\usepackage[russian]{babel}
\usepackage{fp}
\begin{document}
\input{seed}
```

```

\begin{center} Контрольная работа \No 1 \\
Ответы. Вариант \FPprint{n} \end{center}
\FPrandom\x \FPrandom\y \FPrandom\z
\FPrandom\p \FPrandom\q \FPrandom\r
\FPrandom\xx \FPrandom\yy \FPrandom\zz
\FPrandom\pp \FPrandom\qq \FPrandom\rr
\FPeval\x{round((1+10*x):0)}
\FPeval\y{round((1+10*y):0)}
\FPeval\z{round((1+10*z):0)}
\FPeval\p{round((1+10*p):0)}
\FPeval\q{round((1+10*q):0)}
\FPeval\r{round((1+10*r):0)}
\FPeval\xx{round((1+10*xx):0)}
\FPeval\yy{round((1+10*yy):0)}
\FPeval\zz{round((1+10*zz):0)}
\FPeval\pp{round((1+10*pp):0)}
\FPeval\qq{round((1+10*qq):0)}
\FPeval\rr{round((1+10*rr):0)}
\FPeval\pp{clip(pp+p)} \FPeval\qq{clip(qq+q)}
\FPeval\rr{clip(rr+r)}
\FPeval\xx{clip(xx+x)} \FPeval\yy{clip(yy+y)}
\FPeval\zz{clip(zz+z)}
\noindent{\bf Задание 3}.
Номинальный ВВП 1995 г. равен
\FPeval\gdpa{clip(p*x+q*y+r*z)} \FPprint{gdpa},
номинальный ВВП 1996 г. равен
\FPeval\gdpb{clip(pp*xx+qq*yy+rr*zz)}
\FPprint{gdpb},
реальный ВВП 1996 г. равен
\FPeval\rgdp{clip(p*xx+q*yy+r*zz)} \FPprint{rgdp},
дефлятор равен
\FPeval\defl{round((gdpb/rgdp*100):1)}
\FPprint{defl}\%,
инфляция равна \FPeval\infl{round(((gdpb/rgdp-
1)*100):1)} \FPprint{infl}\%, темп роста реального
ВВП равен \FPeval\gr{round(((rgdp/gdpa-1)*100):1)}
\FPprint{gr}\%.
\end{document}

```

Заметьте, что в файле ответов key.tex дублируются команды, генерирующие исходные данные. Важным для генерации случайных

входных данных является тот факт, что использование одного и того же значения для инициализации генератора псевдослучайных чисел (команда `\FPseed`) приводит к одной и той же последовательности псевдослучайных чисел, которые генерируются командами `\FPrandom`. Поэтому последовательность вызовов команд `\FPrandom` должна быть одной и той же для файла с текстом задания и файла с текстом ответов к заданию! Если генератор псевдослучайных чисел инициализировать другим значением, например, 3, что соответствует новому варианту контрольной работы, это приведет к генерации нового набора входных данных.

Можно создать текст ответов к заданиям контрольной работы в том же файле *test.tex* с текстом самой работы, например, на новой странице. Во избежание путаницы при генерации случайных данных, желательно использовать уникальные имена переменных, используемых в рамках пакета `fr`, внутри одного файла.

Пакетный файл *z_pdf.bat* предназначен для автоматической генерации нескольких вариантов контрольных работ при работе с издательской системой TeX в операционной системе Windows.

В строке 1 инициализируется цикл по переменной i , которая принимает значения от 1 до 100 с шагом 1 (будет создано 100 контрольных работ!). В строках 2-3 создается новый промежуточный файл *seed.tex*, содержащий две команды пакета `fr`:

```
\FPset\n{i} \FPseed=i
```

Вместо переменной i подставляется числовое значение, присваиваемое при выполнении цикла. Переменная n , присваиваемая командой `\FPset` отвечает за номер варианта контрольной работы, а команда `\FPseed` инициализирует генератор псевдослучайных чисел значением i (это число должно быть различным для разных вариантов контрольных работ, иначе будут созданы одинаковые наборы входных данных!).

В строках 4 и 6 запускается скрипт *pdflatex* для преобразования текста контрольной работы из файла *test.tex* в файл формата *pdf*. Полученный файл *test.pdf* перемещается с помощью команды `move` в файл с уникальным именем, содержащим номер варианта, в данном случае *test_i.pdf*, где вместо переменной i подставлено числовое значение, присваиваемое при выполнении цикла.

В строках 5 и 7 аналогичные действия проводятся для файла *key.tex*, который содержит ответы на задания контрольной работы. В

строке 8 после выхода из цикла промежуточный файл *seed.tex* удаляется с диска.

Текст пакетного файла *z_pdf.bat*

```
for /L %%i in (1,1,100) do (      1
echo \FPset\n{%%i} > seed.tex    2
echo \FPseed=%%i >> seed.tex     3
pdflatex test.tex                4
move test.pdf test_%%i.pdf        5
pdflatex key.tex                  6
move key.pdf key_%%i.pdf )       7
del /Q seed.tex                   8
```

Следующий скрипт *z_pdf* выполняет аналогичную работу по автоматической генерации набора контрольных заданий для оболочки *bash* в Unix-подобных операционных системах.

Текст скрипта *z_pdf*

```
#!/bin/bash
for ((i=1; i<=100; i++));
do
echo \\\FPset\\n\\{$i\\} > seed.tex
echo \\\FPseed=$i >> seed.tex
pdflatex test.tex
mv test.pdf test_$i.pdf
pdflatex key.tex
mv key.pdf key_$i.pdf
done;
rm seed.tex
```

Вместо команды *pdflatex* для подготовки файлов формата *pdf* или *ps* можно воспользоваться командами *latex*, *dvips* и *dvipdfm*, которые доступны в дистрибутивах MikTeX и TeTeX издательской системы TeX.

В результате обработки файла контрольной работы *test.tex* и текста ответов *key.tex* пакетным файлом были получены следующие тексты.

Контрольная работа №1

Вариант 1

Задание 3. В экономике производится только три товара: апельсины, автомобили и компьютеры. По данным, приведенным в таблице, рассчитайте реальный и номинальный ВВП в 1996 г., считая 1995 год базовым, дефлятор ВВП, инфляцию и темп роста реального ВВП в 1996 г. по отношению к 1995 г.

Годы	1995г.		1996г.	
	Цена	Кол-во	Цена	Кол-во
Апельсины	10	1	16	10
Автомобили	5	8	13	9
Компьютеры	6	8	7	10

Контрольная работа №1
 Ответы. Вариант 1

Задание 3. Номинальный ВВП 1995 г. равен 98, номинальный ВВП 1996 г. равен 347, реальный ВВП 1996 г. равен 205, дефлятор равен 169.3%, инфляция равна 69.3%, темп роста реального ВВП равен 109.2%.

Проблемы при использовании пакета `fr`

Пакет `fr` не будет корректно работать, если вы используете стиль `multido` в `LaTeX`, поскольку в стилевом файле `multido.sty` определены собственные команды `\FPadd` и `\FPsub`. Для этого замените в файле `multido.tex` команду `FPadd` на `mdo@FPadd` и команду `FPsub` на `mdo@FPsub`.

Если используется французский язык пакета `babel`, то возникают проблемы с применением функций `fr`, которые используют символ двоеточия «:». В этом случае пакет `fr` следует подключать в `LaTeX` до пакета `babel` с опцией `french`.

Список литературы

1. Описание и дистрибутив пакета `fp`. – Режим доступа: <http://www.ctan.org/tex/latex/fp>.
2. Издательская система TeX. Дистрибутив MikTeX. – Режим доступа: <http://www.miktex.org>.
3. Издательская система TeX. Дистрибутив TeTeX. – Режим доступа: <http://www.ctan.org>
4. LaTeX Project Public License. – Режим доступа: <http://www.latex-project.org/lppl.txt>

Оглавление

Предисловие.....	3
Базовые функции пакета <code>fp</code>	5
Уравнения, константы, функции.....	7
Вычисление выражений.....	9
Использование пакета <code>fp</code> для генерации заданий.....	11
Проблемы при использовании пакета <code>fp</code>	16
Список литературы.....	17

Учебное издание

Величко Андрей Сергеевич

**ПОДГОТОВКА УЧЕБНЫХ
ЗАДАНИЙ В FLOATING POINT**

Учебно-методическое пособие

В авторской редакции

Подписано в печать 24.09.2007.

Формат $60 \times 84 \frac{1}{16}$. Усл. печ. л. 1,16. Уч.-изд. л. 1,00.

Тираж 50 экз.

Издательство Дальневосточного университета
690950, г. Владивосток, ул. Алеутская, 56

Отпечатано на множительной технике ИМКН ДВГУ
690950, г. Владивосток, ул. Октябрьская, 27

ДЛЯ ЗАМЕТОК

