

Московский авиационный институт
Факультет прикладной математики
Кафедра вычислительной математики и программирования

ПРАКТИКУМ

по циклу дисциплин

ИНФОРМАТИКА

УЧЕБНОЕ ПОСОБИЕ ДЛЯ СТУДЕНТОВ I КУРСА

с приложением хрестоматии на DVD-ROM

Под редакцией к.ф.-м.н., доц. Зайцева В.Е.

МАИ
2007

Авторы:

Зайцев В.Е., Горлов А.А., Дзюба Д.В., Журавлева Т.Э., Измайлов А.А., Калинин А.Л., Крылов С.С., Лебедев А.В., Лукашевич С.Ю., Никулин С.П., Овечкис А.Г., Перетягин И.А., Сеницкий П.А., Сошников Д.В., Титов В.К., Чернышов Л.Н.

Предлагаемый практикум по циклу дисциплин «Информатика» предназначен для студентов первого курса факультета прикладной математики МАИ, изучающих предмет в объеме 14 часов в неделю в течение года. Пособие содержит описание лабораторных работ и заданий курсовых работ и проектов по дисциплинам цикла, программы курсов, включая вопросы к зачетам и экзаменам, а также необходимые справочные сведения о технических и программных средствах.

Также как и теоретический курс, практикум носит фундаментальный университетский характер и может быть полезен для самообразования с целью углубления теоретических знаний и развития практических навыков.

Практикум проводится с 1976 года согласно идее доц. С.С. Гайсаряна. Ранние варианты практикума разрабатывались с участием многих преподавателей кафедры: Борисова Е.В., Луговой И.З., Марасанова А.М., Ротаниной М.В., Трасковского В.Д. и Юдина С.М. С тех пор практикум претерпел значительные изменения и дополнения, его объем вырос втрое. В связи с бурным развитием вычислительной техники и информатики практикум и пособия к нему ежегодно обновляются.

Данное пособие целесообразно использовать совместно с учебником по курсу и с прилагаемой хрестоматией на компакт-дисках, которая рассчитана на самостоятельную работу студентов и содержит многочисленные оригинальные и свободно-распространяемые системы, статьи, документы и материалы, а также различные программные системы, используемые в данном практикуме.

Издание 17-ое, переработанное и дополненное

© Московский авиационный институт, 1979, 1983.

© Московский авиационный институт, 1993–2007, изменения и дополнения

Структура практикума в I семестре 2007/08 учебного года

Практикум состоит из курсового проекта, курсовой и лабораторных работ, практических занятий, консультаций и предполагает самостоятельную работу студентов в объеме не менее 50% планового времени аудиторных занятий по курсу. На I курсе обучение по родственным специальностям идёт по единым календарным планам. Общий недельный бюджет времени студента – 54 академических часа по 45 мин. Нормативный объем самостоятельной работы студентов по дисциплинам цикла информатики определен действующим учебным планом в размере 120 астрономических часов по 60 мин. Академический час для двух- и четырёхчасовых аудиторных занятий в МАИ согласно решению Учёного Совета института сокращён с 45 мин. до 37 мин. 30 с. Для самостоятельной работы студентов, не имеющих домашних компьютеров, в терминальном классе имеется в среднем 12 свободных рабочих мест сверх средней численности группы (16).

Практические занятия (32 часа) проводятся в аудиториях для всей группы возможно с применением компьютера и других технических средств (проектора, ноутбука, ...).

Консультации в группах по курсовому проекту, курсовой и лабораторным работам (105-175 часов на группу 15–25 студентов, включая сдачу зачетов, тестирование программ и проверку отчетов преподавателем в дополнительное время или в режиме *offline*) и лабораторные работы (64 ч.) в подгруппах проводятся в аудиториях и в терминальном классе ОС UNIX (438 ГУК «Б»).

Курсовой проект, курсовая и лабораторные работы должны быть выполнены, оформлены и сданы в объявленные сроки. В осеннем семестре 2007/08 уч. года – до 21 декабря.

В первом семестре по практикуму учебными планами предусмотрено два академических зачета с оценкой (КП и КР) и один без оценки (Практикум /Архитектура ЭВМ и системное ПО).

Студенты, не сдавшие курсовой проект, курсовую работу или зачёт по практикуму (в том числе все лабораторные работы и промежуточные зачеты – контроль знаний по заданиям КП, КР и ЛР) не допускаются к экзаменам по *всем предметам*.

Темы практических занятий

ВВЕДЕНИЕ В ИНФОРМАТИКУ

1. Обработка сообщений. Кодирование, примеры кодов. Тренировка вручную. Перекодировка с помощью программных средств.
2. Системы счисления. Перевод целых и дробных чисел из одной системы счисления в другую, в том числе с кратными основаниями. Представление числовых и текстовых данных в ЭВМ. Тренировка в переводе и представлении. Подготовка к I зачёту.

АЛГОРИТМИЧЕСКИЕ МОДЕЛИ

3. Машины Тьюринга со специализированными командами. Подготовка к лабораторной работе № 5.
4. Диаграммы Тьюринга. Подготовка к лабораторной работе № 6.
5. Нормальные алгоритмы Маркова. Подготовка к лабораторной работе № 7.

Факультативно:

- α. Клеточные автоматы Цузе.
- β. Машины Поста.

ПРОГРАММИРОВАНИЕ (на лекциях и практических занятиях рассматриваются конструкции двух языков программирования, Паскаля и Си; решение задач в группе производится только на одном из них по выбору преподавателя).

6. Структура программы на языках Паскаль и Си. Тип целый: переменные, константы, выражения. Оператор присваивания. Простейший вывод в выходной текстовый файл. Подготовка к лабораторным работам № 8-9.
7. Тип логический: переменные, константы, выражения. Оператор присваивания. Условный оператор. Принадлежность точки области на плоскости. Цикл с предусловием. Приемы отладки программ средствами входного языка. Особенности реализации на языке Си. Подготовка к лабораторной работе № 9.
8. Тип литерный: переменные, константы, выражения. Оператор присваивания. Связь с целым типом. Цикл с постусловием. Простейший ввод из входного текстового файла. Дублирование текстового файла. Подготовка к лабораторной работе № 11 и к выполнению V задания КП.
9. Анализ и преобразование чисел в различных системах счисления. Подсчет числа цифр. Проверка на палиндром. Генерация числового результата преобразования. Подготовка к лабораторной работе № 12.
10. Тип вещественный: переменные, константы, выражения. Оператор присваивания. Индуктивное присваивание и программирование итерационных алгоритмов решения уравнений. Подготовка к выполнению III и IV заданий КП.
11. Процедуры и функции. Передача параметров. Процедура численного интегрирования с параметром-функцией на Паскале и Си. Подготовка к выполнению IV задания КП.
12. Тип диапазон и перечислимый тип. Оператор многозвенного ветвления. Множества. Примеры задач: сигналы светофора, жесты регулировщика, манёвры корабля, санитарный день в библиотеке, решето Эратосфена, подсчет числа букв, цифр, и знаков препинания во входной строке, правила переноса русских слов и др. Особенности реализации на языке Си. Подготовка к лабораторной работе № 13.
13. Циклы с параметром. Массивы. Обработка векторов и строк. Проверка строки на палиндром.
14. Вложенные циклы. Многомерные массивы. Обработка матриц. Ввод-вывод матриц. Сумма элементов выше (ниже) диагоналей. Обмен (удаление) строк и столбцов. Подготовка к лабораторным работам № 14-15.

15. Комбинированные структуры данных. Записи с вариантами и объединения. Оператор присоединения. Оператор варианта для записи с вариантами. Квалификаторы и селекторы доступа к полям структур. Печать внутреннего представления данных.
16. Обзор экзаменационных задач.

Факультативно:

- Υ. Математическое издательство в системе T^EX. Вёрстка отчёта по КП/КР в среде LaTeX по готовому шаблону.
- δ. Системы аналитических вычислений и математических расчетов.
- ζ. Системы научной графики и визуализации.

Темы групповых консультаций

1. Аппаратные средства вычислительного комплекса и терминального класса. Лабораторная работа № 1.
 2. Операционная среда программиста в ОС UNIX на ЭВМ Digital Alpha. Лабораторная работа № 2. Подготовка к I зачёту.
 3. Сети и телекоммуникации в ОС UNIX: TCP/IP, Internet, Intranet. Лабораторные работы № 3-4.
 4. Имитаторы машин и диаграмм Тьюринга и алгоритмов Маркова. Лабораторные работы № 5-7.
 5. Системы программирования Паскаль и Си. Интерактивные отладчики. Лабораторные работы № 8 и № 10. Подготовка ко II зачёту.
 6. Архитектуры вычислительных машин. Подготовка к III зачёту.
 7. Дополнительные возможности ОС UNIX (метасимволы, переадресация ввода-вывода, конвейеры, фоновые процессы).
 8. Архитектура процессора Digital Alpha. Представление данных. Подготовка к IV зачёту.
- Дополнительно:
9. Разработка текстовой утилиты – фильтра ОС UNIX. Подготовка к выполнению V задания КП.

Прием зачета по разделу курса «Архитектура ЭВМ и системное ПО» производится поэтапно (I–IV промежуточные зачеты) в рамках консультаций КП/КР на группу. Количество консультаций в связи с увеличением численности групп, необходимостью приема пересдач, а также ввиду доступности справочных и методических материалов может быть сокращено.

Дополнительно в рамках репетиционного экзамена проводится тестирование знаний по задачам экзаменационной программы. Также устраиваются краткосрочные курсы и тренинги по решению экзаменационных задач.

Общий порядок выполнения и сдачи курсовых заданий и лабораторных работ

Курсовые и лабораторные работы выполняются в соответствии с расписанием машинного времени, выделенного группе (4 акад. часа в неделю), с 1 по 16 неделю семестра в ОС UNIX на ЭВМ Digital Alpha в терминальном классе учебной лаборатории УМЦ-8 (438 ГУК «Б»). Каждому студенту предоставляется отдельный дисплей.

На I курсе использование студентами других ЭВМ разрешается **только** для подготовки *отчетов* по КП/КР и ЛР, *текстов* программ и *предварительной* отладки, в целях тренировки и самообразования.

Получив индивидуальное курсовое или лабораторное задание, студент в течение 1–2 недель составляет программу и сценарий выполнения работы и оформляет предварительный отчет по задаче в соответствии с приведенными правилами (предпочтительно на бланке) и проходит процедуру краткого входного контроля знаний в форме, определяемой преподавателем. Затем программа должна быть отлажена в течение двух недель после выдачи задания.

После отладки программы студент самостоятельно тестирует её и предъявляет на тестирование преподавателю.

В случае успешного тестирования его результаты протоколируются, и листинг подписывается преподавателем.

В случае введения электронного документооборота файлы задачи и листинг могут депонироваться в системе с помощью локальной утилиты *depo*. В таких случаях может быть применена электронная подпись студентом и преподавателем вносимых в реестр электронных документов.

Принятые преподавателем файлы с программами (исходные тексты) и тестовыми данными для КП, КР и ЛР должны сохраняться в директории студента в неизменном виде *до конца учебного года (3 июля 2008 г.)*. Необходимо охранять свои программы от несанкционированного использования, закрывая домашнюю директорию от чтения посторонними: бывали случаи предъявления чужих программ преподавателю раньше, чем это делали их авторы! В таких случаях автору приходится доказывать, что он не плагиатор.

По окончании отладки и тестирования программы студент завершает оформление отчета по задаче и предъявляет его преподавателю для оценки по пятибалльной системе. Оценка задания, выполненного или сданного после установленного срока, снижается. Также оценка может быть снижена за недочеты в программе, в оформлении отчета, за несамостоятельное выполнение работы и за многократное предъявление неработоспособной программы на тестирование преподавателю.

Программа, не прошедшая тестирование преподавателя, оценивается неудовлетворительно, и задание считается невыполненным. Если к концу семестра хотя бы одна программа обязательной части практикума остается неотлаженной, или если отчет по ней не оформлен или не сдан, то **весь соответствующий раздел** практикума (курсовой проект, курсовая и лабораторные работы) оценивается неудовлетворительно. Удовлетворительно могут быть

оценены решения, выполненные не полностью или с погрешностями, но при этом реализующие большинство требований задания и прошедшие большую часть тестов. В этом случае отчёт снабжается анализом ошибок и конкретными предложениями по их устранению. Также могут быть засчитаны задания, выполненные с корректно задекларированными подсказками преподавателя и с участием третьих лиц (при условии успешной защиты).

Лабораторные работы

График выполнения лабораторных работ определяется так, что задания выдаются за 1–2 недели до выхода на ЭВМ и отчеты оформляются в течение не более чем двух недель после отладки.

Темы лабораторных работ:

1. Аппаратные средства UNIX-кластера на базе Digital Alpha.
2. Операционная среда ОС UNIX
3. Локальные сети в среде UNIX.
4. *Глобальные сети и телекоммуникации в ОС UNIX (TCP/IP, Internet, Intranet). (Факультативно).*
5. Отладка программ машин Тьюринга в четвёрках в среде имитатора MT **tu4**;
6. Конструирование диаграмм Тьюринга в среде имитатора **jdt2**;
7. Отладка нормальных алгоритмов Маркова в среде имитатора **nam**.

Отладка программ на стандарте языка Паскаль (Си) в среде Compaq/GNU Pascal (Compaq/GNU C):

8. Системы программирования C/Pascal в ОС. Программы типа «Hello, World/Wirth!».

Далее работы выполняются на одном из языков программирования Паскаль или Си (по выбору преподавателя в группе).

9. Итеративная программа моделирования движения по целочисленной сетке.
10. Методы отладки программ (ручной прокруткой, средствами входного языка, с помощью отладчиков dbx или gdb).
11. Обработка входного текстового файла. Лексический анализ текста с использованием диаграмм состояний и переходов.
12. Системы счисления, представление и обработка целочисленных данных.
13. Работа с множествами (Паскаль) или с битовыми шкалами (Си), простейший анализ текста.
14. Вложенные циклы с параметром. Линеаризация (обход) матриц.
15. Обработка матриц.

Дополнительные лабораторные работы:

16. Клеточные автоматы Цузе.
17. Машины Поста.
18. Математическое издательство в системе T_EX (верстка отчета по курсовой работе в T_EX по готовому шаблону; в весеннем семестре будет обязательная для всех работа на верстку математического текста).
19. Системы аналитических вычислений (Mathematica, Maple, MatLab, ...).
20. Системы научной графики, визуализации и геометрического моделирования (gnuplot, MS Visio ...).

Правила оформления отчетов по лабораторным работам

Отчеты по всем лабораторным работам оформляются на листах формата А3 согласно прилагаемому бланку, который может быть напечатан студентом или воспроизведен в рукописной форме. Бланки антисанитарного вида (слишком мелкий шрифт, загрязнённый фон, испачканные, порванные и т.п.) не принимаются. Отчеты по лабораторным работам сдаются преподавателю по мере готовности и подшиваются в скоросшиватель студента с подложкой из плотной бумаги, выполненной согласно образцу:

Лабораторные работы по циклу дисциплин «Информатика» кафедры 806	
Студент:	Фамилия И.О.
Группа:	08-10х, № по списку
Преподаватель:	Фамилия И.О.

№ работы	Дата	Оценка	Подпись преподавателя
1			
...			
13			
...			

Отчет по лабораторной работе в отличие от отчета по заданию курсового проекта должен быть более кратким и выполняется в упрощенной форме. Отчет по каждой работе, как правило, выполняется на отдельном бланке и должен содержать следующие пункты:

1. *Заголовок* (наименование курса, номер работы в семестре и тема). Например: Лабораторная работа №1 по курсу «Архитектура ЭВМ и системное ПО». Тема: «Операционная среда программиста в ОС UNIX».
2. *Цель работы* (изучение и освоение конкретных аппаратных и программных средств, методов и технологий программирования и решения конкретных классов задач с помощью ЭВМ).
3. *Формулировка задачи* (конкретно, включая № варианта и исходные данные).
4. *Оборудование* (указывается конкретная ЭВМ, на которой выполнялась работа, конкретные наименования устройств). Например: 64-х разрядная микро-ЭВМ Digital Alpha (533 МГц) с ОП 1 Гбайт в сети Gigabit Ethernet, с двумя НМД ёмкостью 54 Гбайт Ultra160 SCSI, с терминальными серверами на 32 канала RS-232C [31 терминал, коммуникационная линия связи (модем) и сетевой матричный принтер LA-120], терминал марки vt-340, подключенный к ЭВМ AXP4, соответствует в системе устройству /dev/tty79. В последующих лабораторных работах допускается описывать только изменения и дополнения к этому стандартному перечню (например, Digital Alpha + графопостроитель HP7580). Если предварительная отладка программы производилась на другом оборудовании (домашний компьютер), следует также привести краткие сведения и о нем.
5. *Программное обеспечение* Указывается наименование используемой версии операционной системы (например, Compaq Tru64 UNIX Release 5.1A with TrueCluster 1.5), системы программирования (например, GNU Pascal version 2.1), редактора текстов (например, ted версии от 14.08.92) и т.п. В последующих отчетах по работе в среде UNIX подробно описываются только дополнительные компоненты ПО (tu4, nam, jdt, GNU C++ Version 3.4).
6. *Идея, метод, алгоритм решения задачи* (коротко; можно привести краткое конструктивное текстовое описание, блок-схему, диаграмму, рисунок, чертеж, таблицу, псевдокод, формальные спецификации, условия и ограничения).
7. *Сценарий выполнения работы* (составляется **до начала** лабораторной работы [включая первоначальный текст программы в черновике (возможно на отдельном листке) и данные для тестирования] с необходимыми пояснениями). Сценарий *без* тестов или соображений по тестированию *не* принимается.

Пункты 1-7 отчета составляются строго до выхода на ЭВМ.

8. *Распечатка протокола* (включая листинг окончательного варианта программы, протокол компиляции и тестирования в одном сеансе цельным куском бумаги) *с подписью преподавателя*.
9. *Дневник отладки* должен содержать дату и время сеансов отладки и основные события (ошибки в сценарии и программе, нестандартные ситуации) и краткие комментарии к ним. Оформляется в виде таблицы. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и третьих лиц в написании и отладке программы. *Незадекларированное участие других лиц является контрафактным*.
10. *Выводы* (достигнута ли цель работы, какие знания и навыки работы получены в результате, какая программа и как отлажена, а также замечания автора **по существу** работы; если задание выполнено с недочетами, то указать причины и пути их устранения).

Курсовой проект

Порядок выполнения заданий проекта

На основе поточных и групповых консультаций студент *самостоятельно* изучает объявленные для данного задания вопросы (конкретные сведения об ЭВМ, ОС, СП, входном языке, методах и алгоритмах решения задач данного класса) и в назначенное для группы время (четыре раза в течение семестра вне расписания) сдает промежуточные зачеты с оценкой. Оценка за сдачу зачета после срока снижается. Конкретный вариант задания выдается **только после** сдачи зачета. По усмотрению преподавателя зачет может проводиться в письменной, устной или комбинированной форме, возможно, с использованием ЭВМ, а также, при наличии и взаимном желании сторон, с применением электронной почты или Интернета.

По желанию студента на кафедре можно заказать ксерокопии методических материалов и страниц книг по курсу, произвести распечатку электронных материалов на лазерном принтере по себестоимости ксерокса в множительном центре МАИ. Кроме того, студентам оказываются следующие дополнительные услуги: доступ через носители и по беспроводным каналам к FTP-серверам, Intranet факультета и к Internet. Необходимые для курсовых и лабораторных работ межмашинное копирование между Digital Alpha, локальный доступ к Intranet факультета (включая WWW) и распечатка на матричном принтере (16 листов формата А3 в семестр) в рамках семестровой квоты машинного времени (2880 мин.) осуществляется бесплатно как плановый учебный процесс. Студентам, своевременно не использовавшим эти ресурсы без документально подтвержденных уважительных причин, они предоставляются дополнительно в соответствии с положением о платных услугах МАИ, либо путем отработки на такелажных, экспедиторских и хозяйственных работах.

Отчет по курсовому проекту (работе) окончательно оформляется и сдается преподавателю в конце семестра строго до начала зачётной недели. По курсовым проектам и работам выставляются зачеты с оценкой. Отчет принимается на проверку при условии сдачи в течение семестра всех промежуточных зачетов по программе курсового проекта и

выполнения всех обязательных заданий проекта и лабораторных работ по соответствующему курсу (если только по нему не предусмотрено отдельного зачёта по лабораторным работам) строго в соответствии с заданными преподавателем вариантами.

Проверка отчетов выполняется, как правило, в течение недели и может включать дополнительное тестирование.

График выполнения курсового проекта, курсовой и лабораторных работ по циклу информатики в 1 семестре 2007/08 учебного года

№ задания КП, ЛР	Макс. оценка	Сдача зачета	Отладка задачи	Оформление отчета
1, 2, 3, 4*, 5, I	5	4 неделя	5 неделя	7 неделя
	4	5 неделя	6 неделя	8 неделя
	3	6 неделя	7 неделя	9 неделя
6, 7, 8, 9, 10, II	5	5 неделя	7 неделя	9 неделя
	4	6 неделя	8 неделя	10 неделя
	3	7 неделя	9 неделя	11 неделя
III, IV, 11, 12	5	7 неделя	9 неделя	11 неделя
	4	8 неделя	10 неделя	12 неделя
	3	9 неделя	11 неделя	13 неделя
V*, 13, 14, 15	5	9 неделя	11 неделя	13 неделя
	4	10 неделя	12 неделя	14 неделя
	3	11 неделя	13 неделя	15 неделя
Отчеты по КП, КР и ЛР в целом	5	14 неделя		
	4	15 неделя		
	3	16 неделя		

Сроки выполнения заданий в конкретных группах могут быть скорректированы преподавателями в связи с отпуском, болезнью, командировкой, неисправностью ЭВМ, праздничными днями, стихийными бедствиями и т.д., о чём делается соответствующая отметка в ведомости текущей успеваемости на кафедре и в лабораторном журнале.

Состав обязательных заданий (необязательные отмечены *) корректируется по решению кафедры.

Вопросы к зачету по разделу курса «Архитектура ЭВМ и системное программное обеспечение»

Часть I

ПРЕДСТАВЛЕНИЕ ДАННЫХ В ЭВМ (10/10/2007)

1. Коды ASCII и КОИ-8: характеристики, состав, структура.
2. Альтернативная и основная кодировки для ПЭВМ (в сравнении с ASCII).
3. Кодировки ISO 8859-5 и CP 1251.
4. Понятие о кодах EBCDIC, ДКОИ.
5. Кодировки Unicode, UTF-8. ISO 8859-5 и 10646.
6. Понятие о клавиатурных раскладках. Основные принципы.
7. Раскладки QWERTY и ЙЦУКЕН и соответствие между знаками кириллицы и латинского алфавита для обычной и фонетической латино-кириллических раскладок.
8. Раскладки Дворак и Diktor.
9. Позиционные системы счисления.
10. Представление целых чисел в ЭВМ.
11. Перевод чисел из одной системы счисления в другую.
12. Особенности целочисленной арифметики в ЭВМ.
13. Научная (экспоненциальная) форма записи числа. Машинное представление.
14. Различия представлений числовых и текстовых данных в ЭВМ.
15. Использование калькуляторов ОС UNIX (bc) и MS Windows для операций с числами в различных системах счисления.

ОПЕРАЦИОННАЯ СИСТЕМА (24/10/2007)

1. Понятие о программном обеспечении ЭВМ. Состав, структура, многоуровневый иерархический характер.
2. Понятие об операционной системе.
3. Краткая историческая справка об ОС UNIX.
4. Особенности ОС UNIX (режим работы, инструментальные и сервисные средства, язык Си, TCP/IP, Internet и пр.).
5. Основы интерактивной работы в ОС UNIX. Начало и окончание сеанса. Доступ к оперативной документации.
6. Команды ОС UNIX. Формат команд. Интерпретатор команд.
7. Протоколирование сеансов ОС UNIX.
8. Файловая система ОС UNIX (структура, именование, навигация).
9. Метасимволы для генерации имен файлов.
10. Управление файлами (создание, копирование, сцепление, переименование, удаление).
11. Управление каталогами (создание, просмотр, удаление).
12. Переадресация ввода и вывода в ОС UNIX.

Часть II

РЕДАКТИРОВАНИЕ ТЕКСТОВ

1. Понятие о редакторе текстов. Основные типы редакторов текстов (строчный, экранный, графический, пакетные и текстовые процессоры).
2. Строчное редактирование. Концепция. Пример: основные возможности стандартного строчного редактора ОС UNIX **ed**.
3. Экранное редактирование. Концепция. Примеры экранных редакторов ОС UNIX (**vi**, **vim**, **emacs**, **ted**).
4. Краткая историческая справка по редактору **emacs**.
5. Базовые возможности **emacs** (переход между кириллицей и латинским алфавитом, запуск и приостановка редактора, выход с сохранением всех измененных буферов и без сохранения, переход в командную строку, завершение имен, откат и перерисовка).
6. Меню редактора: вызов, использование и отмена меню. Виды меню. Структура главного меню.
7. Перемещение курсора в **emacs**.
8. Удаление текста в **emacs**.
9. Работа с блоками текста в **emacs**.
10. Поиск и замена в **emacs**.
11. Работа с буферами и окнами в **emacs**.
12. Операции с файлами в **emacs**.

ОС UNIX

1. Конвейеры команд.
2. Фоновые процессы: запуск, остановка, слежение.
3. Приостановка и возобновление **emacs**. Вызов интерпретатора команд UNIX из **emacs**.
4. Удаленные команды **login**, **who**, **sh** и **uptime**.
5. Команды идентификации узла сети, системы, пользователя, сеанса и терминала.
6. Эмуляция терминала удаленной ЭВМ (**lat**, **telnet**, **PuTTY**).
7. Удаленное копирование и передача файлов и директорий (**rsh**, **rftp**, **kermi**, ...).
8. Безопасность удаленных команд (**scp**, **sftp**, **ssh**, ...).
9. Доступ к файловым системам сети с помощью NFS.
10. Кластеризация ЭВМ в сети ОС UNIX.
11. Управление очередью печати.
12. Команды **dd**, **tr** и **iconv** и их использование для перекодировки файлов.
13. Формат текстового файла в различных ОС. Способы преобразования к формату UNIX.
14. Электронная почта в ОС UNIX. Адресация абонентов. Приёмы использования стандартного почтового клиента.
15. Архивация файлов и директорий с помощью **tar**. Обслуживание архива **tar**: просмотр оглавления, выборка, разархивация.
16. Сжатие/распаковка файлов с помощью **gzip**.

Часть III

СИСТЕМЫ ПРОГРАММИРОВАНИЯ (31/10/2007)

1. Определение, структура и состав СП.
2. Языковые процессоры: трансляторы (компиляторы) и интерпретаторы.
3. Языковая среда.
4. Краткие сведения о GNU проекте. Особенности СП GNU Compiler Collection C/Pascal.
5. Многоязычная СП Tru64 UNIX. Особенности СП Compaq C/Pascal.
6. Компиляция программ в СП C/Pascal.
7. Редактирование связей и создание выполнимых программ.
8. Компиляция с созданием выполнимой программы.
9. Задание флагов СП в командной строке и в тексте программы. Флаги «только компиляция», «компиляция-линкидж», «линкидж», «именование выполнимой программы», «подключение библиотек», «уровень оптимизации», «генерация защитного кода», «трассировка работы компонент СП», «генерация отладочного кода», «проверка стандарта языка», «версия СП».
10. Файлы СП C/Pascal.
11. Понятия отладки и тестирования.
12. Отладка путем ручной прокрутки.
13. Отладка средствами стандарта языка.
14. Отладчики СП GNU и Tru64 UNIX.

ЭВМ (07/11/2007)

1. Понятие об ЭВМ. Процессор, память, внешние устройства.
2. Принципы фон Неймана. Гарвардская и принстонская архитектуры.
3. Структурная схема ЭВМ класса mainframe (сервер). Примеры.
4. Структурная схема ЭВМ открытой шинной архитектуры. Примеры.
5. Процессор. Регистры (команд, данных, адреса). УУ и АЛУ.
6. Понятие о различных архитектурах процессоров (CISC, RISC, VLIW, векторно-конвейерная, транспьютеры, DSP).
7. Выполнение программы в ЭВМ. Взаимодействие процессора (АЛУ и УУ) и памяти.
8. Запоминающее устройство. Память. Слово. Адрес. Бит. Байт.
9. Устройства ввода-вывода. Примеры.
10. Устройства внешней памяти. Примеры.
11. Устройства межмашинной связи. Примеры.

Часть IV

ЭВМ Digital Alpha (14/11/2007)

1. Историческая справка об ЭВМ Digital (PDP-11, VAX-11, microVAX II).
2. Краткая характеристика ЭВМ семейства Alpha.
3. Структура и состав распределенной вычислительной системы МАИ и факультета №8 (центральные, периферийные и сетевые устройства).
4. Особенности архитектуры ЭВМ Digital Alpha.
5. Конкретные характеристики ЦП ЭВМ Alpha. Разрядность. Регистры. Слово состояния.
6. Конкретные характеристики ОП ЭВМ Alpha. Адресуемое пространство. Понятие о реальной и виртуальной памяти.
7. Конкретные характеристики аппаратно реализованных целых типов DEC Alpha.
8. Конкретные характеристики аппаратно реализованных вещественных типов DEC Alpha.
9. Конкретные реализации типов char, real и integer в СП C/Pascal на Digital Alpha.
10. Краткая характеристика ОС ЭВМ семейства Alpha (OpenVMS, Digital UNIX/Compaq/HP Tru64, Linux, FreeBSD, Windows NT).
11. Понятие о других 64-разрядных процессорных архитектурах (MIPS, UltraSPARC, Itanium, PowerPC, HP PA, AMD Opteron). Сравнение с Intel x86 и другими 32-разрядными архитектурами.

Литература к практикуму

1. Бауэр Ф.Л., Гооз Г. Информатика. -М.: Мир, 1976, 1990. Глава 1 и приложения А; Глава 6 и приложение В (D)
2. Ворожук А.В. Основы ЦВМ и программирование. -М.: Наука, 1978. с. 23-39, 59-80.
3. Форсайт Дж., Малькольм М., Моулер К. Машинные методы математических вычислений. -М.: Мир, 1980, с. 22-26
4. Карасев С.Б. и др. Машинные алгоритмы обработки информации. -М.: МАИ, 1987. с. 3-16.
5. Гайсарян С.С. и др. Средства организации программ сложной структуры. -М.: МАИ, 1982. с. 3-25.
6. Калянов Г.Н. CASE – структурный системный анализ. –М.: Лори, 1996. с. 9–18, 46–56, 111–116.
7. Экранный редактор текстов etacs. ШПАРГАЛКА пользователя. -М.: МАИ, 1997. - 2 с.
8. Бек Л. Введение в системное программирование – М.: Мир, 1988. с. 28–33, 123–138, 145–146, 156–162, 222–223, 264–265, 294–296, 299–302, 305–307, 312–364, 386–389, 411–429.
9. Беляков М.И. и др. Мобильная операционная система. -М.: Радио и связь, 1991. с. 155-157, 161-164.
10. Цикритзис Д., Бернштейн Ф. Операционные системы. - М.: Мир, 1977.
11. Краткое руководство по GNU Pascal. -М.:МАИ, 1996. - 4 с.
12. Танненбаум Э. Многоуровневая организация ЭВМ. -М: Мир, 1979
13. Шадский А., Архитектура Alpha AXP// Открытые системы, 1993, No 4, с. 32–33, 44–48.
14. Digital Technical Journal, Vol. 4, No. 4, Special Issue 1992.
15. Малые ЭВМ высокой производительности. /Г.П. Васильев и др. -М.: Радио и связь, 1990, с.3-14, 251. Ц. 1р. 20к.
16. Сибеста Р. Структурное программирование на языке ассемблера ЭВМ VAX-11. - М: Мир, 1988, с. 52-71, 73-78, 172, 224, 254-257, 352, 369, 461.
17. UNIX on-line manuals (rcc, rlogin, rwho, telnet, rsh, ruptime, hostname, ftp).
18. Компьютер Уорлд Москва. 1993, NN 11,19, Компьютер Уик Москва, N 30, 1996, с. 43.
19. Компьютеры. Справочное руководство в 3-х томах. Под ред. Г. Хелмса. -М.: Мир, 1986.
20. Грубов В.И. и др. Справочники по ЭВМ. -Киев: Наукова думка, 1989.
21. Мячев А.А., Степанов В.Н. Персональные ЭВМ и микроЭВМ. Основы организации. -М.: Радио и связь, 1991, с.94.
22. Перспективы развития вычислительной техники Кн. 5. Малые ЭВМ / Н.Л. Прохоров и др. -м.: Высшая школа, 1989.
23. Levy H.M., Erkhous R.H., Computer Programming and Architecture The VAX. Digital Press, 1989.
24. Дейтел Г. Операционные системы. - М.: Мир, 1987. Том 2, тт. 16.1,2,3,10,11,12, гл. 18,19.
25. Кристиан К. Введение в операционную систему UNIX. -М.:Финансы и статистика, 1985.
26. Баурн С. Операционная система UNIX. -М.:Мир, 1986.
27. Using and Porting the GNU Pascal compiler, Version 2.1 for GCC 2.95. – 373 с. (ксерокопируется по заказу)
28. GNU Pascal On-Line Manual (Доступно на AXPI/AXP2 и CD-хрестоматии).
29. Официальный сайт проекта GNU: <http://www.gnu.org>
30. Архив программного обеспечения Simtel: <http://www.simtel.net/>
31. Официальный сайт проекта GNU Pascal: <http://www.gnu-pascal.de/>
32. Официальный сайт проекта Free Pascal: <http://www.freepascal.org>
33. Готье, Руководство по ОС UNIX, –М.: Финансы и статистика, 1985.
34. МакМаллен Дж., UNIX (Complete Idiot's Guide) // Пер. с англ. –М.: ЮНИТИ, 1996.
35. Браун П., Введение в операционную систему UNIX, –М.: Мир, 1987.
36. UNIX. Руководство системного администратора. –К.: BHV, 1997.
37. Банахан М., Раттер Э. Введение в операционную систему UNIX. –М.: Радио и связь, 1986.
38. Керниган Б.В., Пайк Р. UNIX – универсальная среда программирования. –М.: Финансы и статистика, 1992.
39. Инструментальные средства разработки программ в современных операционных системах // Учебное пособие под ред. С.М.Юдина. - М.: Изд-во МАИ, 1990, с. 38-53, 78-85 (интерпретатор команд cshell).
40. Мейер Б., Бодуэн К. Методы программирования: в 2-х томах. –М.:Мир, 1982.
41. Храмов П., Лабиринты Интернет. - М., 1995.
42. Топхем Д., Хай Ван Чыонг. Юникс и Ксеникс: Пер. с англ. – М.: Мир, 1988.
43. Тихомиров В.П., Давидов М.И. Операционная система ДЕМОС: инструментальные средства программирования. – М.: Статистика, 1988. (C Shell, make, lex, yacc).

44. Йенсен К., Вирт Н. Паскаль. Руководство пользователя и описание языка. – М. 1980.
45. Абрамов С.А. и др. Задачи по программированию. – М.: Наука, 1988. Зайцев В.Е., Сошников Д.В. и др. CD-хрестоматия по курсу Информатика. – М.: 2002 – в 2-х тт. на CD-ROM. Кнут Д. Искусство программирования для ЭВМ. Т 1. Основные алгоритмы. – М.: Мир, 1976.
46. Жоголев Е.А. Лекции по технологии программирования. – М.: ИО ф-та ВМиК МГУ, 2001. (ISBN 5-89407-008-2) Главы 2,10. см. также <http://www.cmc.msu.ru>
47. Compaq Pascal Online Manual(man pc).

Правила оформления курсового проекта (работы)

Одной из важнейших целей курсового проекта является привитие студентам навыков правильного оформления научно-технических отчетов, программной и технической документации в соответствии со стандартами ЕСПД, ЕСКД и НТО.

Отчет оформляется в скоросшивателе, обложка которого должна иметь следующий вид:

ПРАКТИКУМ	
по циклу дисциплин «Информатика»	
курсовой проект, курсовая и лабораторные работы	
Студент:	Фамилия И.О.
Год приема:	2006
Группа:	08-10х
Руководитель:	Фамилия И.О.

В скоросшивателе следует собрать отчеты по курсовому проекту, курсовой и лабораторным работам 1 семестра. Текст отчета оформляется на односортной нелинованной писчей бумаге формата А4 (210×297 мм) плотностью 50–80 г/м². Допускаются отклонения от размера не более 7×9 мм. В сшитом отчете должен быть обеспечен свободный доступ к каждому листу и листингу, т.е. **не разрешается** использовать прозрачные конверты для кластеризации фрагментов отчета.

Написание текста производится с одной стороны листа от руки, четко и аккуратно; допускается печать на пишущей машинке или принтере через 1.5 интервала шрифтами 12 кегля. Листы нумеруются в правом верхнем углу. Номер 1 присваивается первому листу, следующему за титульным (бланку задания). Цифра 1 на нем не ставится.

В тексте отчета сокращения слов не разрешаются (за исключением общепринятых, например, т.к.). Допустимы аккуратные исправления, заклеивание или закрапка.

Первый лист отчета — титульный (ему не присваивается номер) — отделяет данный КП (КР) от других отчетов. Он выполняется на плотной бумаге и должен иметь вид, приведенный ниже.

Московский авиационный институт (государственный технический университет)	
Факультет прикладной математики	
Кафедра вычислительной математики и программирования	
КУРСОВАЯ РАБОТА	
по курсу	
"Информатика"	
I семестр	
«Составление, отладка, тестирование и документирование программ на языке Си в среде ОС UNIX»	
Студент:	Фамилия И.О.
Группа:	08-10х., № по списку

Дж. Фон Нейман	А.П. Ершов	Д. Э. Кнут	Д. С. Скотт	А.Л. Пажитнов
А. Тьюринг	В.А. Мельников	Н. Вирт	Б.Керниган	Тим Бернерс-Ли
Д. Ритчи	К. Томпсон	Е.А.Жоголев	М.Р. Шура-Бура	Р. Флойд
В.П. Иванников	С.С. Гайсарян	Э. В. Дейкстра	Дж.Бэкус	Р.В. Хэмминг
Ч. А. Р. Хоар	М. Минский	Э.Ф. Кодд	А.Ньюэлл, Х.Саймон	Ф.Дж. Корбатто
Ч. Бэббидж	А. Лавлейс	К. Цузе	А.А. Марков	Э. Фейгенбаум

С.А. Лебедев	Л.Н. Королев	Э. Пост	Дж.Буль	Ф. П. Брукс, мл.
К. Шеннон	Л.Торвальдс	Р. Столлманн	О.Б. Лупанов	К. Нигард
С.С. Лавров	Г.С. Цейтин	Я. Лукашевич	Б. Страуструп	О-Дж. Даль
В.Э. Вольфенгаген	В.Ф. Хорошевский	Л.Г. Райков	В.Ш. Кауфман	Дж. Хопкрофт
А.С. Кронрод	В.К. Титов	А.Н. Терехов	Е.М. Ландис	В.Л. Арлазаров
Г.М. Адельсон-Вельский	А.А. Ляпунов	Б. Паскаль	Б. Мейер	Л. Эллисон

См. также сайты с биографиями <http://www.thocp.net/biographies/biographies.htm> и <http://www.biography.com>. Биографии Б. Гейтса и П. Нортон к рассмотрению не принимаются.

Пример списка терминов словаря

Бит	Процессор	Устройства ввода/вывода	Операционная система
Байт	Арифметическое устройство	Устройства межмашинной связи	Файловая система
Регистр	Устройство управления	Машинное ε	Программное обеспечение ЭВМ
Слово	Запоминающее устройство	Интерпретатор команд	Система программирования
Адрес	Устройства внешней памяти	Компиляторы и интерпретаторы	Языковая среда
Отладка	Тестирование	Прокрутка	Редактор текстов

- *отчеты по заданиям*, оформленные в соответствии с правилами, приведенными ниже;
- *заключение* (2-3 стр.), содержащее сведения о достижении поставленных целей, о завершенности отладки программ, а также оценку сложности и полезности изученного материала и замечания и предложения студента **по существу** данного проекта;
- *список использованных источников* (1-2 стр., литература, документация, лекции, консультации, оперативная документация, материалы на магнитных и оптических носителях, BBS, FIDONet, ftp- и WWW-серверов и др.). Наименования книг приводятся по образцу приведенных выше ссылок;
- *приложения* (заверенные преподавателем распечатки текстов программ и протоколов тестирования, полученные в учебной лаборатории факультета, **цельными кусками**).

Заголовки разделов записываются более крупными буквами в верхней части первого листа раздела. Все разделы должны начинаться с новой страницы.

Отчет по отдельному заданию курсового проекта включает следующие подразделы:

1. *постановка задачи*: полная формулировка условия задачи с указанием номера варианта;
2. *общий метод решения* (с кратким обоснованием);
3. *общие сведения о программе*: необходимое программное и аппаратное обеспечение, операционная система, язык и система программирования, число строк программы; *местонахождение и имена* файлов с исходными текстами и данными, способ вызова и загрузки;
4. *функциональное назначение*: для выполнения каких вычислений предназначена программа, ограничения на объем и величины обрабатываемых данных.
5. *описание логической структуры*: словесное описание алгоритма, блок-схема или псевдокод;
6. *описание переменных и констант* в виде таблицы

<i>Имя</i>	<i>Тип</i>	<i>Назначение</i>
i	integer	вспомогательная переменная (счетчик цикла)
N	const integer	размер исходного массива
M	array [1..N, 1..N] of real	исходная матрица

7. Если программа имеет подпрограммы, то функциональное назначение, описание логической структуры и описание переменных и констант приводятся отдельно для каждой процедуры или функции. При описании функционального назначения подпрограммы дополнительно приводятся имя, вид и тип возвращаемого значения (если есть) а также информация о формальных параметрах в виде таблицы:

function fact(x : integer): integer;

Функция, вычисляющая факториал целого числа

<i>Имя</i>	<i>Тип</i>	<i>Вид</i>	<i>Назначение</i>
x	integer	вх. параметр. (по значению)	входной параметр

i	integer	лок. перем	счетчик цикла
fact	integer	возвращаемое значение	результат вычисления – факториал

8. *входные данные*: характер, организация, предварительная подготовка и формат входных данных;
9. *выходные данные*: характер, организация и формат выходных данных;
10. *тестовые примеры*: исходные данные для нескольких различных сценариев тестирования программы и ожидаемые результаты, либо соображения по характеру тестовых данных.
11. *дневник отладки* (дата, время и место (если использовались другие ЭВМ) основных событий отладки, время, затраченное на отладку, наиболее характерные ошибки, их внешние признаки и способы локализации и исправления, сведения о степени самостоятельности выполнения работы);
12. *выводы по задаче* (достигнута ли цель задания, какие знания и навыки работы получены в результате выполнения, а также замечания автора **по существу работы**; если задание выполнено с недочетами, то объяснить причины и указать пути их устранения).

Особенности оформления отчётов по КП(КР), не связанных с программированием, приведены в соответствующих заданиях.

Зачеты по практикуму

Зачет без оценки («Архитектура ЭВМ и системное ПО» или «Практикум на ЭВМ») выставляется за выполнение, оформление, защиту и сдачу лабораторных работ, предусмотренных обязательным перечнем, студентам, успешно прошедшим 4 этапа промежуточного контроля знаний. Дополнительные лабораторные работы могут быть использованы для повышения оценки по практикуму, не заменяя при этом ни одну из обязательных работ или заданий.

Зачет с оценкой по КП (Архитектура ЭВМ/Практикум. на ЭВМ) выставляется по результатам выполнения заданий I-II, с учетом оценок за лабораторные работы №№ 1-3, 4*, 8 и 10 и промежуточные зачеты I-IV.

Зачет с оценкой по КР (Информатика, АЯП) выставляется по результатам выполнения заданий III-IV-V*, с учетом оценок за лабораторные работы № 5-7, 9, 11-15 и пояснительную записку (реферат, ...).

Олимпиады по программированию и математике

Для студентов первого курса проводятся олимпиады по программированию: в декабре и в мае <http://www.mai.ru/projects/microsoft/olimp.html>. С середины I семестра организуются еженедельные очные тренинги для подготовки к олимпиадам с последующим разбором задач и их заочным дорешиванием (www.amais.org.ru). Сначала рекомендуется в заочном или очно-заочном режиме самостоятельно выполнить квалификационное задание в домашних условиях или в лаборатории. В результате самостоятельного выполнения квалификационного задания за ограниченное время студент может убедиться в своей способности участвовать в очных турах олимпиады.

Правила проведения очных туров олимпиады соответствуют принятым на чемпионатах мира ACM, но с личным первенством. Кроме того, задачи формулируются на русском языке. Участникам предлагается набор задач для решения на ЭВМ в течение 5 часов с автоматическим тестированием на наборе тестов жюри. Победители и призеры олимпиад получают премиальные баллы к экзамену и право номинироваться в резерв сборной команды МАИ для подготовки и участия в межвузовских матчах, чемпионате Москвы, четверть- и полуфиналах чемпионата мира, награждаются грамотами и подарками МАИ и Microsoft, удостоиваются премий и именных стипендий.

Пример олимпиадного задания (Задача «Биллиард», автор Титов В.К., 1983 г.) см. в задании к ЛР №9. Тесты и правильные ответы к нему будут даны на пробном туре.

Студенческие олимпиады по математике освещаются на странице http://www.mai.ru/colleges/fac_8/math_olimp/index.htm

Студенческий список рассылки stud@shwarsico.lists.ru

Вы можете принять участие в дискуссиях по актуальным вопросам науки и образования, в том числе в области информатики, программирования и вычислительной техники применительно к специфике факультета с участием студентов, аспирантов и преподавателей кафедры, подписавшись на почтовый список рассылки stud@shwarsico.lists.ru. Отправленное по этому адресу письмо рассылается всем подписчикам; подпоследовательности таких писем образуют целенаправленные нити дискуссий в жанре публичной переписки.

Подписаться на список рассылки можно одним из следующих способов:

- послав письмо с просьбой администратору списка рассылки по адресу studlist@shwarsico.com
- автоматически со страницы <http://shwarsico.lists.ru/mailman/listinfo/stud/>
- автоматически при помощи почтового робота, направив письмо с командой *subscribe* по адресу stud-request@shwarsico.lists.ru.

После этого Вы будете регулярно получать сообщения из списка рассылки (помеченные в теме письма подстройкой [Stud] – обычно не более 1-2 сообщений в день). Обратите внимание, что по умолчанию ответы на такие письма также рассылаются **всем** абонентам списка.

Отказаться от подписки можно одним из трех способов, обратных к подписке (т.е. *unsubscribe*).

Архивы списка рассылки с марта 2000 г. находятся по адресу <http://shwarsico.lists.ru/pipermail/stud> - прежде, чем подписаться, Вы можете ознакомиться с происходившими в прошлом дискуссиями и обменами мнений.

В настоящее время список рассылки насчитывает более 300 подписчиков.

Форум для студентов 8 и 18 факультетов по информатике и смежным вопросам

<http://faq8.mailabs.ru>

Разделы: абитуриенты, 1-й курс, учёба, дипломники, аспиранты, выпускники, II высшее образование, олимпиады, информатика, переводники и др.

Часть V (принимается в зачётную неделю или в 2007 г.)

ОС UNIX. ОСНОВЫ АРХИТЕКТУРЫ.

1. Режим разделения времени ОС UNIX.
2. ОС UNIX: ядро и утилиты.
3. Различные оболочки ОС UNIX (интерпретаторы команд shell, cshell, kshell, bash, ...).
4. Мультипрограммирование: свопинг и виртуальная память.
5. Процессы ОС UNIX.
6. Начальная загрузка ОС UNIX.
7. Периферийные устройства ОС UNIX.
8. Понятие о программировании на ИКЯ (Shell, Cshell, bash...): переменные, параметры, выражения, присваивание, проверки, ветвления, циклы.

КОМПЬЮТЕРНЫЕ СЕТИ. ИНТЕРНЕТ.

1. Основы построения компьютерных сетей. Понятие протокола. Протоколы различных уровней.
2. Семейство протоколов TCP/IP как основа построения локальных и глобальных сетей на базе ОС UNIX.
3. Доменная система имен. Сеть интернет.
4. Основные сервисы в рамках интернет: ftp, telnet, email и т.д.
5. Понятие гипертекста и гипермедиа. Основные принципы построения всемирной паутины WWW.
6. Понятие о языке разметки гипертекстовых документов HTML.
7. Использование в сети интернет различных платформ и операционных систем. Понятие о гетерогенных компьютерных сетях.

Бильярд представляет собой клеточный прямоугольник $M \times N$. В клетке с координатами (I, J) находится шар. В переменной K задано значение, определяющее направление по диагонали, в котором начинает двигаться шар. Направление кодируется следующим образом: 1 - вправо-вверх, 2 - вправо-вниз, 3 - влево-вниз, 4 - влево-вверх. Шар движется по диагонали до стенки бильярда и отражается от нее, продолжая движение по перпендикулярному диагональному направлению. Движение шара заканчивается выходом из бильярда, если он попадает в одну из его угловых клеток (лунку).

Требуется по заданным M, N, I, J, K определить, выйдет ли шар за край бильярда, и если выйдет, то через какой угол и за сколько ударов о стенку бильярда. Если шар не выйдет за пределы бильярда, надо проследить его движение до попадания в начальную точку с начальным направлением движения и подсчитать число ударов в одном цикле траектории.

Составленная программа должна обработать несколько тестов вплоть до конца файла.

Исходные данные: В каждой строке входного файла задается отдельный тест:

5 целых чисел, разделенных пробелами: четыре 2-значных числа и одно 1-значное. Исходные данные корректны:

$M, N \geq 2; 1 \leq I \leq M; 1 \leq J \leq N; 1 \leq K \leq 4$.

Результат для каждого теста должен содержать число ударов о стенку и название угла, если шар попадает в лунку.