

Министерство образования и науки Российской Федерации
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ

Кафедра «Сверхвысокочастотной и квантовой радиотехники»

Д.Г. Осетров, Ю.А. Павличенко, Н.Д. Хатьков, Л.И. Шангина

**Руководство для подготовки обучающего и контролирующего
комплекса компьютерных методических разработок при организации
самостоятельной работы студентов**

Учебное пособие

2009

Содержание

Введение	2
1. Основные возможности и свойства системы Sydney	4
2. Техническое описание системы “Sydney” и ее возможностей	5
3. Технология изготовления методического обеспечения в Sydney	6
4. Создание модулей Sydney	8
5. Создание мультимедийных лекций	10
6. Создание мультимедийных лабораторных работ. Программа SLab	15
7. Изготовление тестового материала.	25
Литература	37

ВВЕДЕНИЕ

Самостоятельная работа студентов является неотъемлемой частью учебного процесса подготовки квалифицированных специалистов, способных самостоятельно и творчески решать, стоящие перед ними задачи. Самостоятельная работа - способ активного, целенаправленного приобретения студентом новых для него профессиональных знаний и умений без непосредственного участия в этом процессе преподавателей. Самостоятельная работа формирует профессиональные знания студента.

Правильная организация самостоятельной работы студентов имеет особо важное значение для студентов всех видов обучения: дневного, заочного и вечернего обучения.

Самостоятельная работа студентов определяется образовательным стандартом «Государственные требования к минимуму содержания и уровню подготовки выпускника».

К видам самостоятельной работы, предлагаемым данным руководством относятся:

- систематическое чтение литературы (в этот раздел входят электронные учебники, созданные преподавателями по своим дисциплинам);
- выполнение индивидуальных расчетных заданий, заданных в тестовом или текстовом исполнении;
- Использование INTERNET при подготовке к семинарским, практическим занятиям, позволяет студентам расширить кругозор и познакомиться с новейшими научными открытиями;
- подготовка к лабораторным занятиям (натурным и виртуальным), которые обязательно включают входной и выходной контроль;
- самостоятельное изучение некоторых вопросов учебной программы, для которых подготавливаются мультимедийные ресурсы или делается ссылка на список литературы.

По перечисленным критериям наиболее перспективной для изготовления комплекса методических материалов является предлагаемая ниже система мультимедийного обучения Sydney – авторами, которой являются Хатьков Н.

Павличенко Ю. Инструментальная основа - лицензионное программное обеспечение Delphi 4.0.

Состав у обучающегося - библиотеки курсов, изготовленных преподавателями при помощи инструментария Sydney.

1.Основные возможности свойства системы Sydney

- Предлагаемая система обучения позволяет эффективно изучать учебные курсы, даже если компьютер находится дома. Это обеспечивается следующими свойствами данной системы:
- Основной объем учебной информации преподносится в естественной звуковой форме. Звуковое сопровождение несет информацию по учебным курсам к какому-либо видеоряду – как на обычной лекции. Во время лекции выделяется достаточный объем времени, чтобы студент мог записать важные определения, содержащиеся в текстовых вставках с формулами. В это же время, когда студент записывает формулы и определения, может звучать легкая приятная музыка, снимающая нервное напряжение во время обучения.
- Важным достоинством предоставления учебной информации в звуковой форме является показ правильного произношения сложных технических терминов, чего лишен студент, занимающийся по заочной форме обучения.
- Немаловажным является и то, что задействуются практически все виды памяти – моторная при записи лекций, слуховая при их прослушивании и зрительная при визуальном просмотре видео информации. Это способствует прочному закреплению в памяти пройденного материала.
- После каждого академического часа (45 минут) устанавливается перемена на 5 или 10 минут, во время нее прокручиваются слайды красивых мест природы и звучит классическая музыка. Студент также по своему желанию может в любой момент остановить изучение материала.
- Во время изучения учебного материала в интерактивном режиме осуществляется текущий контроль знаний студента посредством его опроса в тестирующем модуле. Если же студент не усвоил материал, система на основе результатов его тестирования определяет автоматически те темы, который студент плохо изучил и предлагает их снова до тех пор, пока не будут получены удовлетворительные результаты. Только после получения положительной оценки по пройденному материалу можно будет изучать следующий учебный материал.
- Для текущей оценки успешности изучения студентом учебного материала у каждого студента имеется свой накапливаемый рейтинг, который он всегда видит.

- Студент всегда имеет возможность выбора того или иного курса для его изучения в течение любого времени.
- Над составлением методического материала могут работать преподаватели любой дисциплины.

2. Техническое описание системы “Sydney” и ее возможностей

- Программное обеспечение легко устанавливается, как в локальной компьютерной сети любого компьютерного центра, так и на домашнем компьютере. Оно совместимо с широко-распространенными операционными системами Windows 98/NT. Вследствии применения современных технологий сжатия информации, заложенных авторами в программное обеспечение, на одном CD.
- Отличительная особенность системы - построение по модульному принципу не только в технической реализации, но и в методических материалах. Поэтому система имеет открытую архитектуру и может наращивать свои ресурсы.
- Встроенные программные инструментальные средства, совместимые с широко используемыми текстовыми и графическими редакторами, позволяют в короткие сроки создавать большие объемы методического материала.
- Применение администрирования внутри системы и рейтинговая оценка знаний, также позволяет иметь полную оперативную информацию о каждом обучаемом, что очень важно для администрации образовательного учреждения

В системе Sydney реализованы все возможности для проведения самостоятельного обучения. Это:

- лекционный модуль с поддержкой как двумерной (2dmm), так и трехмерной (3dmm) графики и анимации, звука, текста,
- экзаменационный модуль, реализующий как тестовую проверку знаний, так и проверку с подсказками (тренажерный режим),
- лабораторный модуль, реализующий лабораторные работы любой степени сложности.

Построитель курсов Sydney предоставляет возможность составлять курс с произвольным количеством модулей в произвольном порядке. При этом есть возможность отправлять студента на повторное изучение материала при неудовлетворительном прохождении тестовых заданий.



Рисунок 2.1 Построение курса.

- Впервые в систему встроен трехмерный редактор озвученных динамических изображений (3D аниматор+звук) и лабораторный стол, содержащий произвольные математические модели, что позволяет использовать при обучении трехмерные изображения и звук – в естественной для восприятия человеком форме. Это существенно повышает наглядность и ускоряет процесс обучения.
- Требуется при минимальном размещении – Pentium - 75 МГц , ОЗУ – 24Мб , звуковую карту Sound Blaster, CD-ROM. Размер места на винчестере зависит от того как используется CD-ROM и может составлять от 20 Мб до 400Мб. Для 3D технологий – Pentium II-III, 128 Мб - ОЗУ

Российским агентством по патентам и товарным знакам (РОСПАТЕНТ) выдано свидетельство об официальной регистрации программы для ЭВМ № 990548

Особенности практической эксплуатации.

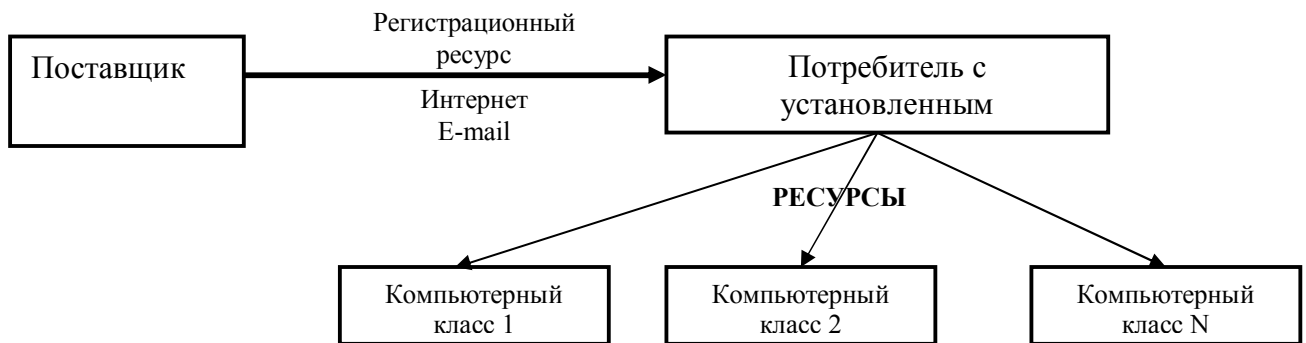


Рисунок 2.2 Связь между обучаемыми и поставщиками программ

- Эксплуатация системы требует наличия электронной почты у потребителя или модемной непосредственной связи с поставщиком для получения расходуемых регистрационных ресурсов. Однако, объем передаваемой информации между потребителем и поставщиком – очень мал и составляет килобайты/мес. Это позволяет использовать только почтовые программы Интернет, а следовательно, тратить очень малые средства на связь.
- Наиболее эффективное применение системы осуществляется в компьютерном классе, обслуживаемом системным администратором.

3. Технология изготовления методического обеспечения в Sydney

3.1. Виды методических материалов:

- Текстовый модуль (ТМ)
- Экзаменационный модуль (ЭМ)
- 2D - мультимедийный модуль (2DMM)
- 3D- мультимедийный модуль (3DMM)
- лабораторный стол, включающий стандартный модуль отчета к лабораторной работе (ЛС)

3.2. Инструментарий для подготовки и проверки электронных методических материалов:

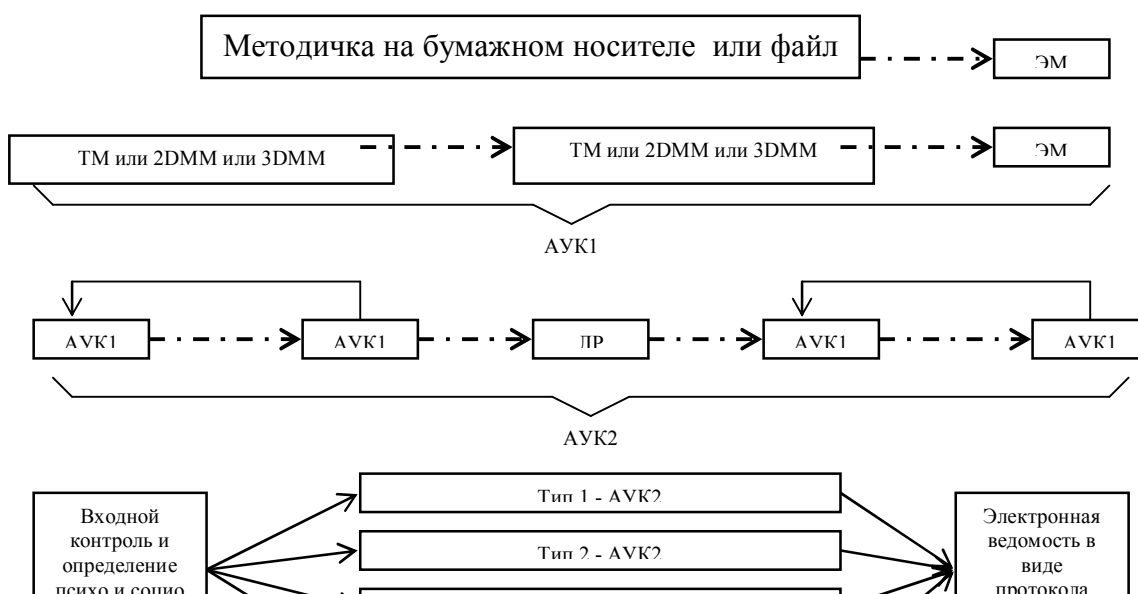
- Мультимедийный дизайнер лекций и описаний – ТМ, 2DMM
- Трехмерный мультимедийный редактор – 3DMM
- Построитель тестов + их отладчик – ЭМ

3.3. Представление методического ресурса - в файловом виде:

- Вся текстовая информация, включающая формулы, графики, рисунки предоставляется в файле текстового процессора **Word** - *.rtf согласно представленным шаблонам или правилам заполнения – ТМ, ЭМ, 2DMM, 3DMM
- Все графические изображения в файлах *.bmp от произвольного графического редактора ТМ, ЭМ, 2DMM, ЛС, 3DMM
- Все математические модели представляются в файлах *.mcd математического процессора **MathCad** и заполненного шаблона.
- Все аудио файлы в формате аудио файла без использования процедур сжатия *.wav произвольного аудио редактора
- Все видео файлы в формате файла без использования процедур сжатия *.avi произвольного видео редактора.

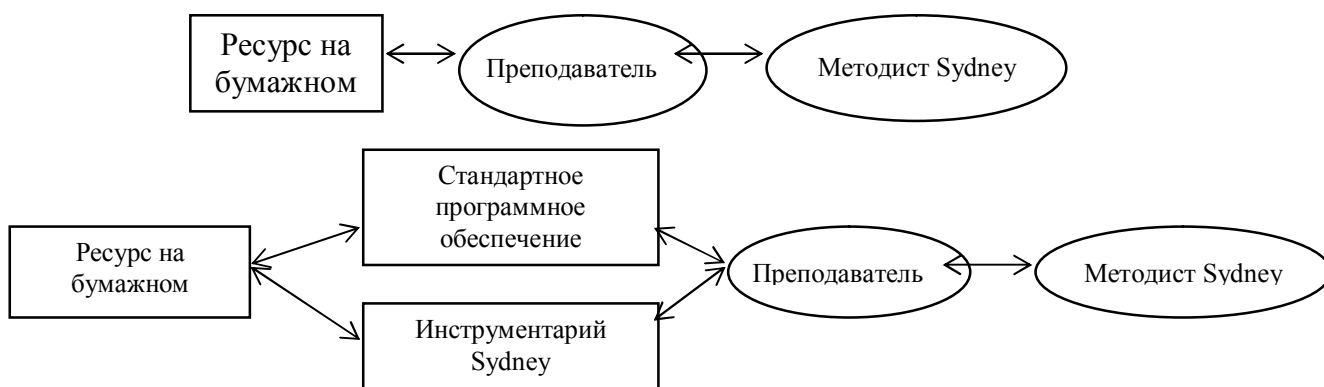
3.4. Компоновка методического материала:

А. Компоновка автоматизированных учебных курсов, поддерживаемых диспетчером курсов (АУК):



3.5. Этапы организации работ по изготовлению методических материалов.

- a) Проведение методических семинаров по использованию инструментальной среды.
- b) Изготовление ресурсов методических материалов преподавателем
- c) Взаимодействие преподавателя с методистом Sydney – приемка метод. материалов согласно техническим требованиям.



- d) Изготовление методических материалов в лаборатории Sydney – регистрация, компиляция, проверка на работоспособность.
- e) Предоставление бета-версии на проверку преподавателю
- f) Взаимодействие с методистом Sydney по исправлению недочетов
- g) Формирование первой версии методических материалов.

4. Создание модулей Sydney:

4.1. Создание текстового модуля (ТМ).

- Для создания ТМ по первому варианту схемы П.5.с) преподавателю достаточно предоставить в рукописном или в уже готовом напечатанном виде методическое пособие и отдать его методисту Sydney.
- Для создания ТМ по второму варианту схемы П.5.с) преподавателю необходимо предоставить файл текстового процессора Word, содержащего необходимую метод. информацию, которая может включать в себя текст, рисунки, графики, формулы в формате указанном П.3. методисту Sydney

4.2. Создание экзаменационного модуля (ЭМ).

- Для создания ЭМ по первому варианту схемы П.5.с) преподавателю достаточно предоставить в рукописном или в уже готовом напечатанном виде экзаменационные вопросы и ответы, согласно рекомендациям экзаменационного шаблона Sydney и отдать его методисту Sydney.
- Для создания ЭМ по второму варианту схемы П.5.с) преподавателю необходимо предоставить либо файл текстового процессора Word, заполненного в соответствии с требованиями экзаменационного шаблона Sydney, который может включать в себя текст, рисунки, графики, формулы в формате указанном П.3. методисту Sydney, либо файл в формате инструментария Sydney, предназначенного для подготовки экзаменов.

4.3. Создание 2D - мультимедийного модуля (2DMM).

- Для создания 2DMM по первому варианту схемы П.5.с) преподавателю необходимо представить методисту Sydney несколько типов рукописных документов – а) текст для озвучивания б) рисунки того, что озвучивается в) схемы перемещения рисунков согласно описанию, если требуется в) сценарий следования кадров.
- Для создания 2DMM по второму варианту схемы П.5.с) преподавателю необходимо предоставить методисту Sydney несколько типов документов а) файл текстового процессора Word, содержащий текст для озвучивания или аудио файлы б) электронные изображения для озвучивания, видео файлы в) сценарий следования кадров, либо один файл в формате инструментария Sydney, предназначенного для подготовки 2DMM.

4.4. Создание 3D- мультимедийного модуля (3DMM).

- Для создания 3DMM по первому варианту схемы П.5.с) преподавателю необходимо представить методисту Sydney несколько типов рукописных документов – а) текст для озвучивания б) объемные рисунки того, что озвучивается в) схемы ракурсов сцены согласно описанию, если требуется в) сценарий следования ракурсов сцены.
- Для создания 3DMM по второму варианту схемы П.5.с) преподавателю необходимо предоставить методисту Sydney несколько типов документов а) файл текстового процессора Word, содержащий текст для озвучивания или аудио файлы б) 3D электронные изображения для озвучивания в) схемы ракурсов сцены согласно описанию, если требуется в) сценарий следования ракурсов сцены,, либо один файл в формате инструментария Sydney, предназначенного для подготовки 3DMM.

4.5. Создание лабораторного стола (ЛС).

- Создание ЛС предполагает только второй вариант схемы изготовления П.5.с)

- Для создания ЛС необходимо предоставить методисту Sydney несколько типов документов а) файл математического процессора MathCad, содержащий математическую модель исследуемого явления б) заполненный шаблон Sydney этой математической модели в) сценарий исследования математической модели г) графические изображения явления исследования с соответствующими обозначениями д) области перемещения изображений (начало-конец) в соответствии с математической моделью, если требуется.

4.6. Создание лабораторной работы (ЛР).

- Преподаватель предоставляет спецификацию в документе Word на последовательность подключения готовых модулей и их взаимодействие - ТМ, 2DMM, 3DMM, ЭМ и ЛС согласно схеме 4.2.
- По мере наращивания методических ресурсов для ЛР, преподаватель может построить достаточно длинную цепь из модулей, охватывающей все аспекты исследуемого явления.

4.7. Создание автоматизированных учебных курсов (АУК).

- Преподаватель предоставляет спецификацию в документе Word на последовательность подключения готовых модулей, в которой указывается автор модулей, организация, версия, рейтинг и др. параметры
- По мере наращивания методических ресурсов, преподаватель последовательно переходит от АУК1 до АУК3.

Смотри в приложении Образец изготовления лабораторной работы

- Создание и анализ графиков.

5. Создание мультимедийных лекций

Designer – на сегодняшний день, пожалуй, одна из лучших программ для создания мультимедийных приложений. Она не обладает таким мощным набором функций, как, например, Macromedia Flash, но очень проста, обладая, тем не менее, всем необходимым для создания таких приложений, как мультимедийные лекции (на создание которых, эта программа, собственно, и была ориентирована) или тесты с запоминающимся, интересным интерфейсом. Отличительная особенность программы – она проста в использовании и не требует от пользователя необходимых в других случаях навыков программирования.

Программа ориентирована на преподавателей средних и высших учебных заведений, использующих в обучении компьютеры, а также на создателей простых, но интересных мультимедиа-приложений.

Работа с Designer

При входе в программу перед пользователем возникает окно с рабочими панелями и собственно рабочим пространством посередине.

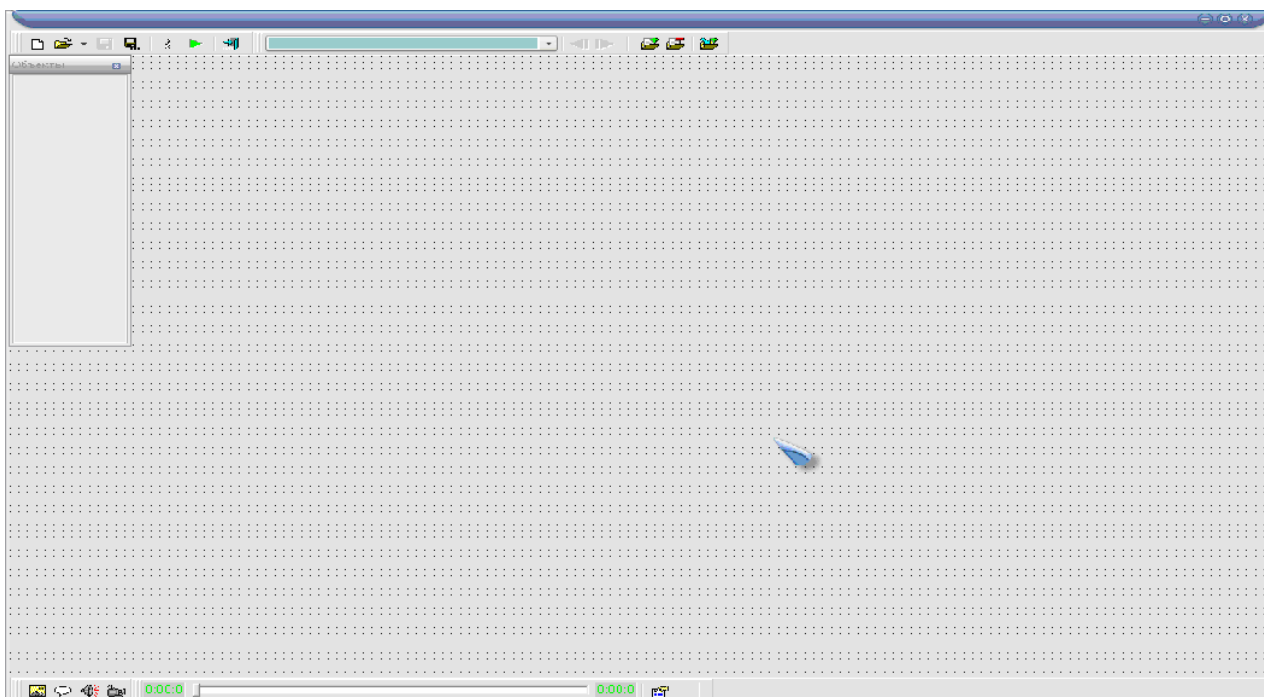


Рис. 5.1 – Общий вид (окно) программы

Вся работа с программой заключается в последовательных осмысленных нажатиях на кнопки на панелях (или выборе функций в контекстном меню, вызываемом правой клавишей мыши), и изменении используемых в создании конечного файла (называемого модулем) графических/аудио/видео файлов.

Модули (те ролики, которые получаются в итоге), делятся на кадры (примерно таким же образом, как фильмы).

Рабочие панели


На панелях окна программы расположены следующие иконки:

 создать новый модуль

Данная команда создаёт новый модуль (являющийся обычным файлом с расширением .slm), который пользователю, собственно, и предстоит «препарировать». После нажатия данной иконки Designer открывает новое окно, и, если пользователь уже успел поработать, просит сохранить модуль. (См. команду «сохранить модуль»).

 открыть модуль

При нажатии иконки «открыть модуль» Designer открывает окно, в котором пользователь среди имеющихся файлов на компьютере может выбрать нужный ему (разумеется, файл этот должен иметь расширение .slm, то есть быть исполняемым файлом Designer'а -- модулем).

 сохранить модуль (когда эта функция не может быть активирована, иконка сереет)

Данная функция очень важна. Достаточно сказать, что, не сохранив имеющиеся наработки, пользователь не сможет отыскать нужный ему модуль в будущем – модуля попросту не будет в памяти компьютера. Поэтому, создав свой шедевр, ОБЯЗАТЕЛЬНО нужно его сохранить. При этом достаточно выбрать функцию «сохранить модуль», и, когда Designer выдаст окно сохранения, ввести

имя файла и папку, где этот файл должен лежать. Позже его будет достаточно просто найти. (См. команду «открыть модуль»). Необходимо также упомянуть, что при выходе, если вы всё же не сохранили свой модуль, Designer, будучи довольно умной программой, сам предложит сохранить имеющиеся данные. Также, если в процессе работы вы уже сохранили свои наработки, что рекомендуется делать почаще – Designer уже не будет спрашивать, куда сохранить файл – он просто переписет имеющийся.



сохранить модуль под другим именем

Данная команда используется в основном тогда, когда вы вдруг решили сохранить файл под другим именем. Последовательность действий та же, что и при использовании команды «сохранить модуль».



изменить свойства модуля (название модуля, Ф.И.О. автора модуля, задание первого кадра и другие)

При помощи этой команды можно изменить свойства открытого модуля, например, задать название модуля, Ф.И.О. автора модуля, номер первого кадра (то есть начать можно не с первого кадра модуля, а, например, с 4.), задать уникальный индекс и максимальный рейтинг модуля (необязательно), и включить внутренний проигрыватель. То есть звуковые и видео файлы будут проигрываться не «внешним», обычно используемым проигрывателем медиа-файлов, а «внутренним», заданным программой.



проиграть модуль (т.е. воспроизвести последовательно все кадры, входящие в модуль)

Все кадры, входящие в модуль, будут воспроизведены у вас на глазах. То, что вы увидите, и будет конечным результатом ваших трудов.



выйти из программы

В комментариях не нуждается. Вам всё это надоело, и вы выходите из программы.



переход к предыдущему кадру (здесь функция не активна)

Просто переход к предыдущему кадру.



переход к следующему кадру (здесь функция не активна)

Просто переход к следующему кадру.



добавить кадр (просто кадр, на нём пока ничего нет)

При выборе данной команды в вашем модуле появится ещё один кадр, пока пустой.



удалить кадр (если не нужен)

Кадр исчезнет из модуля. При этом Designer спросит вас, действительно ли вы хотите удалить данный кадр?



добавить кадры из другого модуля

Допустим, у вас есть готовый модуль, а вы на его основе хотите сделать ещё один. В этом случае достаточно выбрать данную функцию. Designer запросит модуль, откуда вы эти кадры хотите взять. Вы выбираете, и затем достаточно будет просто удалить всё ненужное.



добавить в кадр картинку

Добавляет рисунок.



добавить в кадр надпись

Добавляет надпись.



добавить в кадр звук

Добавляет файл с расширением .wav



добавить в кадр видео

Добавляет файл с расширением .avi, .mpg, .mov



просмотреть свойства кадра

Используя данную функцию, можно изменить свойства кадра, то есть задать его имя, цвет фона, минимальный временной отрезок (то есть шаг прокрутки временной дорожки, расположенной внизу), и максимальное время кадра.



использовать координатную сетку (для удобства расположения элементов в кадре)

Нужно напомнить, что те же действия можно выполнить, нажав правую клавишу мыши и выбрав требуемое действие.

Также, дважды щелкнув по рабочему пространству, можно вызвать меню свойств кадра (название, основной цвет фона, максимальное время кадра)

Щёлкнув правой кнопкой мыши на рабочей панели программы, можно выбрать, какие панели нужны пользователю, а какие – нет. Это увеличит рабочее пространство.

При добавлении в кадр картинки и выборе её на панели «Объекты» (которая находится справа на рабочем пространстве программы), видим панель редактирования картинки, на которой расположены следующие иконки:



загрузить картинку из файла



захватить картинку на экране монитора (то же можно сделать, нажав клавишу PrtScr на клавиатуре)



отредактировать картинку в редакторе (по умолчанию – MS Paint)



удалить картинку



поместить картинку на переднем плане



поместить картинку на заднем плане



сделать картинку прозрачной



иконка создания движения (для этого нужно, нажав иконку, выбрать картинку и переместить её в какое-либо место на рабочем столе, затем ввести в секундах время передвижения)

Для надписи последние пять функций аналогичны функциям картинки, первые три означают:



изменение шрифта надписи



изменение цвета рамки надписи



изменение цвета фона надписи

Для звука на панели изображены следующие иконки:



загрузить звук из файла









открыть звуковой файл для редактирования в фонографе











начать запись звукового файла



остановить проигрывание или запись звукового файла

-  проиграть звука
-  в начало звукового файла
-  в конец звукового файла
-  запомнить позицию начала записи (т.е. выбираем начало, отрезав всё лишнее)
-  стереть запись ДО запомненной позиции (здесь – не активна)
-  стереть запись ПОСЛЕ запомненной позиции (здесь – не активна)

Для видеозаписи функции следующие:

-  загрузить видео файл
-  удалить видео файл
-  поместить видео файл на переднем плане
-  поместить видео файл на заднем плане
-  остановить проигрывание видео файла
-  проиграть видео файл
-  в начало видео файла
-  в конец видео файла

Важно заметить, что, щелкнув правой клавишей мыши по объекту (будь то звук, картинка, надпись или видеофайл), можно данный объект выключить, то есть, не удаляя его из кадра, сделать неактивным.

5.1.Создание мультимедийных лекций в Designer

Выше были подробно объяснены основные функции программы, с помощью которых можно ею управлять. Теперь можно приступить собственно к работе с программой.

Для создания мультимедийной лекции необходимо иметь, в первую очередь, электронную версию ваших лекций, то есть лекции в формате MS Word или иных текстовых редакторов. Имея на руках информацию, добавляем в кадры то, что нам нужно. При этом последовательность появления картинок, звуков, видеозаписей и надписей мы можем установить так, как посчитаем нужным, используя временную дорожку кадра. То есть если мы планируем, например, вызвать на экран текст лишь после того, как сыграет мелодия, нам достаточно установить курсор дорожки в то место, где она заканчивается, и вставить надпись или картинку. Точно так же, если нужно, чтобы вслед за текстом появилась картинка, устанавливаем курсор в нужное место, и отключаем текст, щёлкнув правой кнопкой мыши по иконке, изображающей текст, в панели объектов, и выбрав «выключить». Таким образом очень легко установить очерёдность и продолжительность тех или иных событий. Можно, например, по мере того, как играет музыка, заставить картинку двигаться. Для этого достаточно установить курсор на момент, когда вы хотите начать анимацию, установить картинку в нужное положение, а затем, выбрав команду «движение» в панели картинки, переместить её в нужное положение. Designer запросит время, за которое перемещение должно быть совершено.

Таким образом, очерёдность и событий определяется лишь тем временем, которое вы выберете на временной дорожке кадра.

Как видно, ничего сложного в использовании программы нет. Достаточно расположить картинки, видео файлы и надписи в нужных местах, добавить движение, вставить необходимое количество кадров – имеем красивый, законченный мультимедиа файл.

В заключение можно добавить, что, хотя эта программа довольно слаба и по многим параметрам проигрывает тому же Macromedia Flash, она стоит того, чтобы неопытный пользователь обратил на неё внимание и учился создавать файлы, использующие движение, музыку и звук, именно на ней.

6. Создание мультимедийных лабораторных работ. Программа SLab.

Обучение, в котором используется только лекции и тесты без лабораторных работ, будет однобоким. Обучаемый, не увидев реализацию процесса, будь то физический, химический или другой, не освоит преподнесенный ему материал должным образом. Для устранения такого пробела в обучении была создана программа SLab. Её назначение – моделирование лабораторных работ. Программа позволяет наглядно продемонстрировать какой-либо процесс (рис. 6.1).

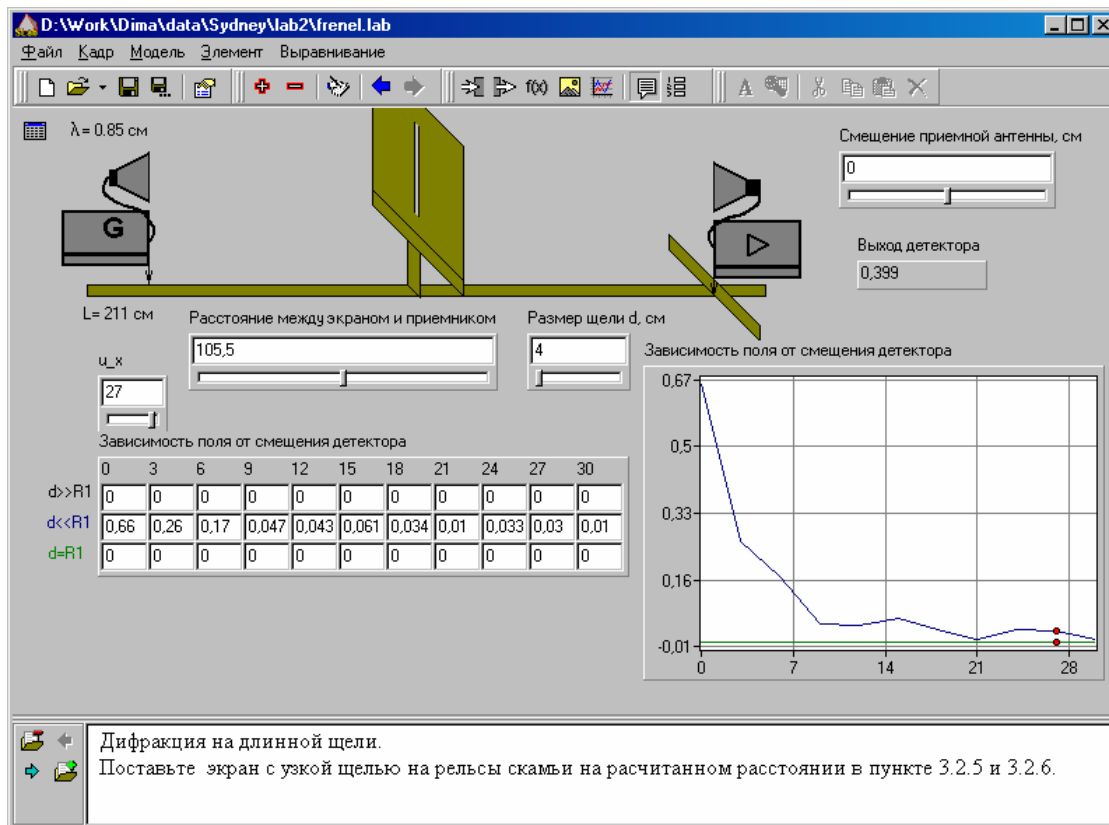


Рисунок 6.1 Лабораторный стол “Дифракция на длинной щели”.

Основные элементы программы, позволяющие вводить данные и выводить полученные результаты, а также повышающие наглядность это:

- числовой ввод,
- логический ввод,
- выбор из списка,

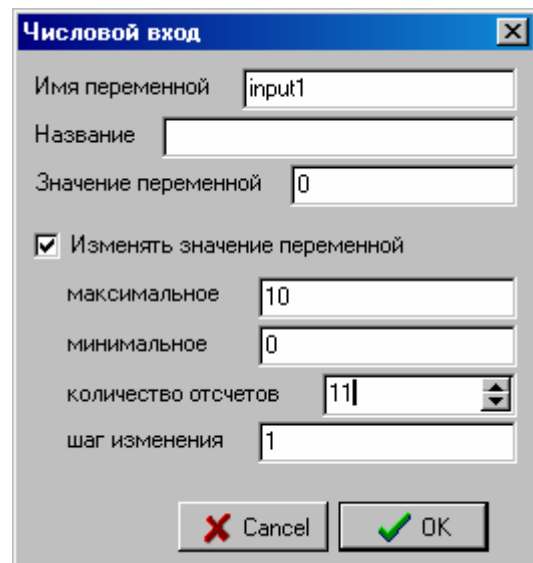
- массив чисел,
- числовой выход,
- выражения,
- функции,
- макромодели,
- двумерный график,
- рисунок,
- текст.

Каждый из элементов (Эта часть выполняется разработчиками системы по написанному сценарию преподавателя) вставляется на лабораторный стол путем выбора пункта меню Модель или нажатием соответствующей кнопки на панели инструментов. При этом программа запросит у пользователя дополнительные данные, такие как имя, под которым будет производиться обращение к элементу, и т.д. Опишем каждый из этих элементов, чтобы преподаватель имел понятие как лучше расположить методический материал, как доступнее представить развитие сценариев эксперимента, как реальнее представить математическую модель физического процесса, как все эти процессы изобразить в графическом исполнении.

Элемент “Числовой ввод”.



а)



б)

Рисунок 6.2 “Числовой ввод” а) Вид элемента, б) Окно ввода данных.

Этот элемент позволяет как вводить данные с клавиатуры, так и выбирать число из диапазона. Границы диапазона, а также шаг изменения величины задается в окне, которое отображается при вставке элемента на лабораторный стол (рис. 6.2.б). Параметр “Название” определяет имя элемента, под которым он будет отображаться на лабораторном столе. Для элемента на рисунке 2.6.а это “Число”

Элемент “Логический ввод”.

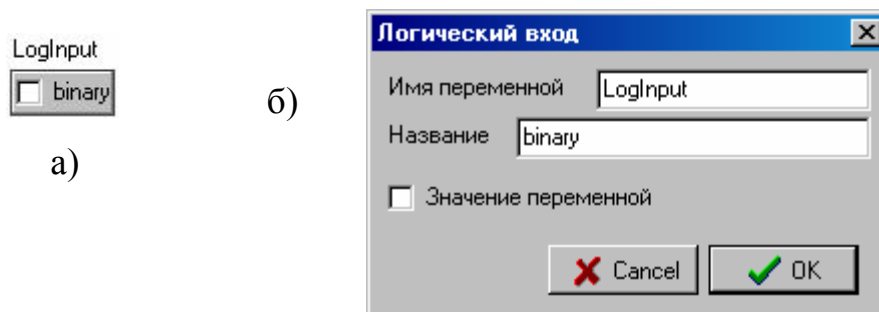


Рисунок 6.3 “Логический ввод” а) Вид элемента, б) Окно ввода данных.

Предназначен для ввода логических данных вида да/нет (1/0). Если отображение имени переменной нежелательно, то оно убирается в контекстном меню Вид | Показывать заголовок.

Элемент “Список”

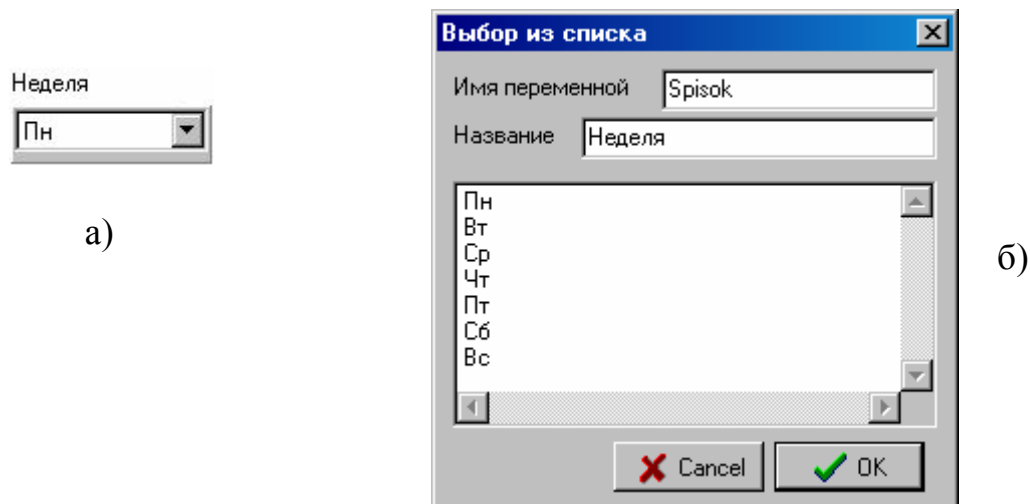


Рисунок 6.4 Элемент “Список” а) Вид элемента, б) Окно ввода данных.

Этот элемент предоставляет возможность выбора одного значения из перечисленного в списке множества значений (Рис. 6.4.б). При выборе значения, переменной присваивается порядковый номер этого значения в списке.

Элемент “Массив чисел”.



а)

б)

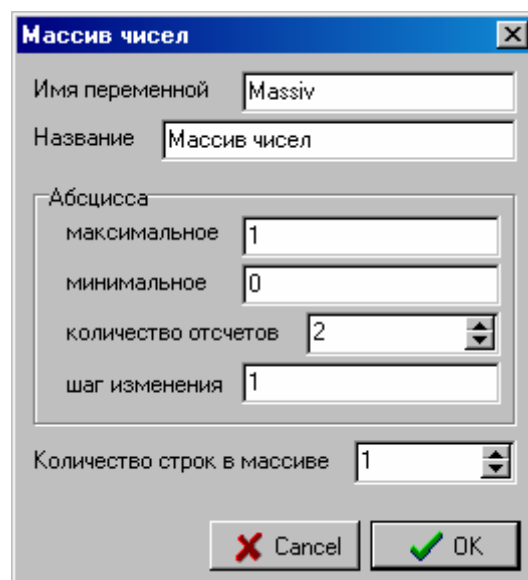


Рисунок 6.5 Элемент “Массив чисел” а) Вид элемента, б) Окно ввода данных.

Основное назначение элемента – предоставление информации в виде одномерной таблицы для построения двумерных графиков. Массив можно сделать двумерным, изменив параметр “Количество строк” в окне ввода параметров элемента. При этом можно будет строить графики для каждой строки таблицы.

Часть элемента, под именем “имя переменной_x”, на рисунке 2.9.а это Massiv_x, предназначен для выбора активной ячейки массива и, по не отображается.

Элемент “Числовой выход”.

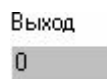
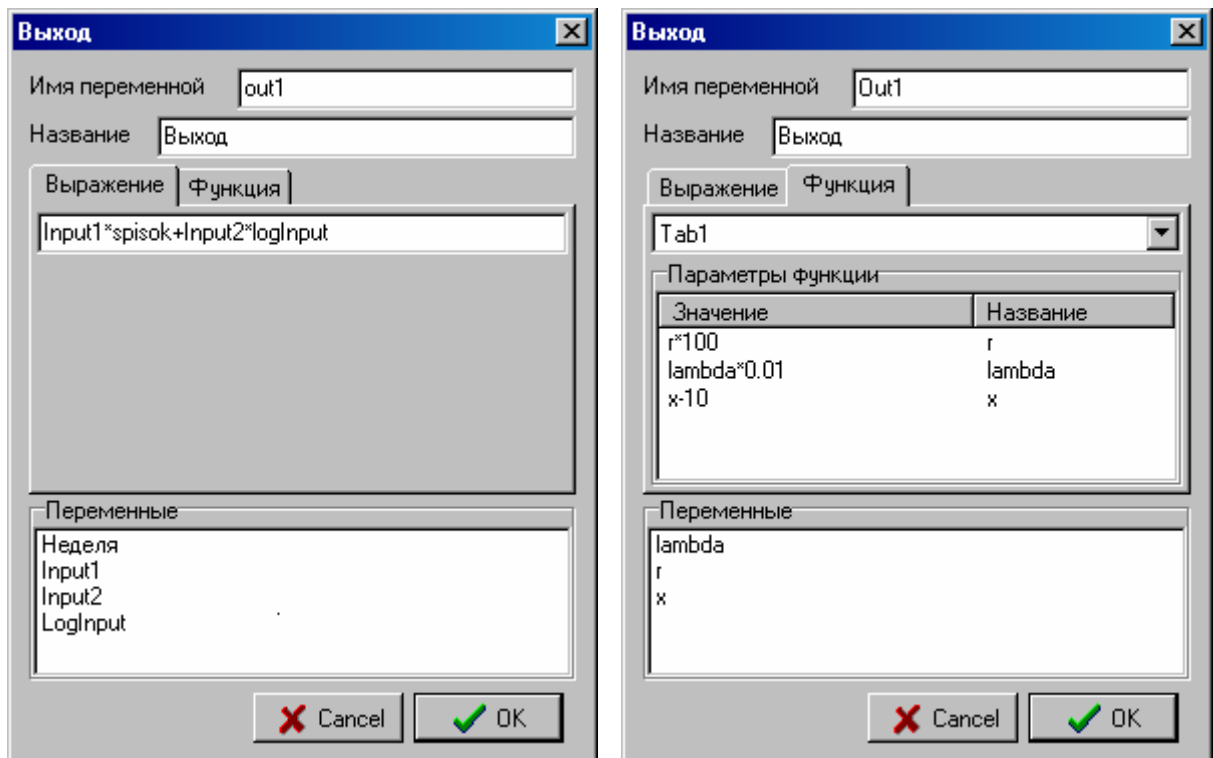


Рисунок 6.6. Вид элемента “Числовой выход”.

Элемент предназначен для вывода числового результата как выражения, так и функции (рис. 6.6). Выражение строится из переменных, перечисленных в нижней части окна ввода данных (рис. 6.7.а). В выражении могут присутствовать только простые арифметические операции (+-*/).



а) б)
Рисунок 6.7. Окно ввода а) выражения, б) функции

Вычисление функций основано на использовании таблиц, содержащих результаты расчета этих функций. Таблицы вставляются путем выбора элемента меню Модель|Добавить функцию|Таблица.

Для построения функции необходимо выбрать из списка таблицу. В области окна ввода данных "Параметры функции", с правой стороны, отобразятся имена формальных параметров, используемых функцией (Рис. 6.7.б). В левой части области вводятся фактические параметры функции, в виде переменных и выражений.

Элемент "Выражение".



а)

б)

Рисунок 6.8 Элемент “Выражение” а) Вид элемента, б) Окно ввода данных.

Этот элемент (рис 6.8.а) дублирует функцию элемента “Числовой выход”- “Выражение”. Построения выражения осуществляется из формальных параметров $u_0, u_1 \dots$ (рис. 6.8.б). Выражение подключается в элементе “Числовой выход” в качестве функции. Это дает возможность использовать одно и то же выражение, но с разными фактическими параметрами.

Элемент “Таблица”.



а)

б)

Вход	минимум	максимум	количес...
r	0,1	2	20
lambda	0,0085	0,0125	9
x	0	0.1	21

Рисунок 6.9 “Таблица” а) Вид элемента, б) Окно ввода данных.

Простые алгебраические выражения не позволяют строить математические модели сложных процессов. Необходимо средство, позволяющее вычислять значения функций, любой степени сложности. Но на просчитывание сложных формул, например содержащих интегралы и дифференциалы, будет уходить недопустимо много времени.

В элементе “Таблица” вычисление функции основано на использовании многомерных таблиц, с размерностью равной числу входных данных. Эта таблица строится в любом математическом процессоре, таких как MathCad, MatLab и имеет вид:

общее число переменных,

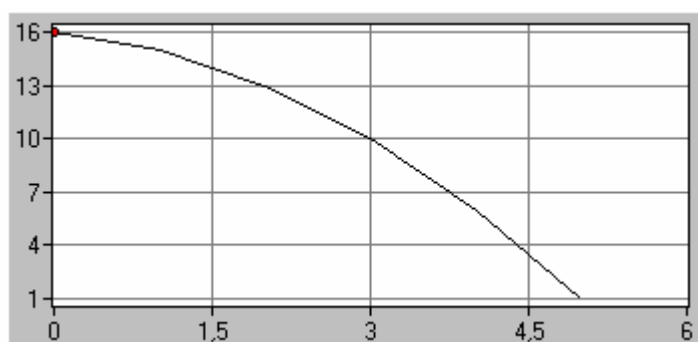
имя первой переменной,
массив значений первой переменной,

...

имя i -той переменной,
массив значений i -той переменной,
массив значений результата функции.

Следует учесть, что программа MathCad не умеет сохранять в файл символьные значения, поэтому в файл таблицы, имена переменных надо будет добавлять вручную в любом текстовом редакторе. Добавление таблицы осуществляется выбором пункта меню Модель|Добавить функцию|Таблица.

Элемент “График”.



а)

б)

Рисунок 6.10 Элемент “Таблица” а) Вид элемента, б) Окно ввода данных.

Элемент может строить от одного до пяти графиков (рис. 6.10.а). Графики строятся на основе данных занесенных в элемент “Таблица чисел”. При построении графика, значением X будет номер столбца таблицы и обозначается как “имя таблицы_x” (рис. 6.10.б). Значением Y_n будут значения элементов строки таблицы, обозначаемой как “имя таблицы_номер строки”. Если не указаны параметры Y_{max} и Y_{min} , то элемент будет автоматически масштабироваться.

Элемент “Рисунок”.

Этот элемент предназначен для повышения наглядности лабораторного макета. Изменяя координаты X и Y (рис. 6.11), можно реализовать движение

картинки по лабораторному столу. Также можно изменять видимость элемента, в зависимости от значения какой либо логической переменной.

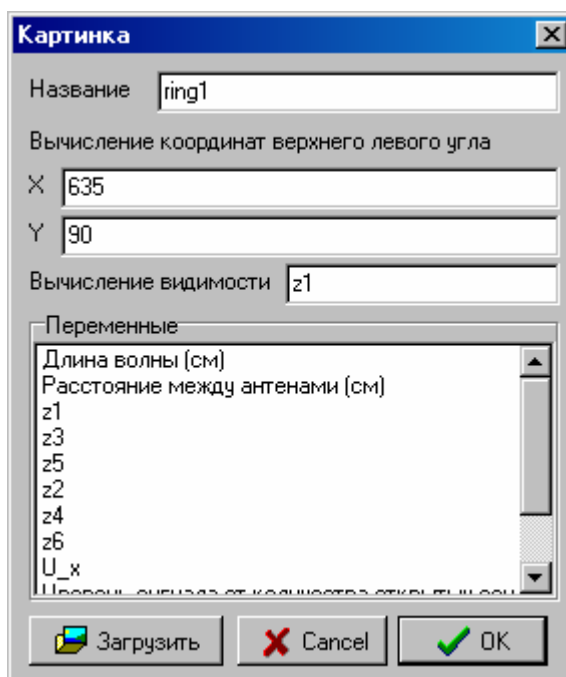


Рисунок 6.11 Окно ввода данных элемента “Рисунок”.

Аналогом методического указания по лабораторной работе является поле инструкций. В этом поле можно поместить в текстовом виде инструкцию, подсказку, т.д. Добавление и удаление инструкций производится кнопками, расположенными на панели инструкций в левом нижнем углу окна программы.

Результатом выполнения лабораторной работы является отчет. В отчет можно вставить любой элемент, если установлена опция “Доступно в отчете” контекстного меню элемента. Перейти к окну редактирования образца отчета можно выбором пункта меню Кадр|Отчет. В левой части окна располагаются все элементы кадра, доступные в отчете. В образце отчета можно использовать системные переменные, вместо которых будут автоматически подставлены соответствующие значения. Например:

- Переменная %НАЗВАНИЕ% - название лабораторной работы,
- Переменная %ФИО% - фамилия и инициалы обучающегося,
- Переменная %ДАТА% - текущая дата,
- Переменная %ГОД% - текущий год.

Приложение:

Образец изготовления лабораторной работы

КОМПЬЮТЕРНАЯ ЛАБОРАТОРНАЯ РАБОТА

Теория

Краткое описание основных теоретических сведений, описание лабораторной установки.

Форма представления:

- текстовый или гипертекстовый документ;
- иллюстрации и звуковое сопровождение.

Входной контроль (итоговый контроль)

Компьютерный тест. Каждая задача теста состоит из вопроса и эталона ответа.

Ответы могут быть следующих типов:

- выбор правильного ответа из нескольких ответов (строк, картинок, формул)
- упорядочивание строк
- указание категории каждой строки из набора строк
- строковый ответ (слово или словосочетание)
- числовой ответ (возможно задание эталона ответа в виде функции)

Форма представления: текстовый документ.

«Лабораторный стол»

Иллюстрации

Структурная схема, чертеж и т.д.

Форма представления: рисунок или графический файл.

Математическая модель исследуемого объекта

Модель представляется в виде функции нескольких параметров и конечных интервалов изменения каждого параметра. Алгоритм составления математической модели:

- Определить количество и состав измеряемых (исследуемых) величин. Изменяемые величины будем называть выходами модели Y . Например, при исследовании полета ядра необходимо измерять две координаты ядра и две проекции скорости ядра.
- Определить количество и состав параметров модели и переменных, от которых зависят измеряемые величины (значения выходов модели). Параметры и переменные будем называть входами модели X . Для каждого параметра и переменной необходимо определить диапазон их изменения. Кроме диапазона желательно задать количество интервалов, на которые будет разбит диапазон изменения. В нашем примере в качестве входов модели выступают начальная скорость ядра, угол стрельбы, ускорение свободного падения, время полета.

Для каждого выхода модели составить математическую функцию $y=f(x_1, x_2, x_3\dots)$ или эквивалентный массив значений, полученных при исследовании реального объекта.

Форма представления: документ MathCAD.

Порядок выполнения работы

Пошаговая инструкция по выполнению исследования.

Форма представления: текстовый документ.

Шаблон отчета

Титульный лист, названия разделов отчета, краткие рекомендации по оформлению.

Форма представления: текстовый документ.

Перечень работ, выполненных преподавателем при разработке мультимедийных лабораторных работ в системе «SYDNEY»

1. Разработка методического описания и ввод в машину лабораторной работы:
2. Разработка входного и выходного контрольных вопросов и ввод в машину
3. Разработка сценария лабораторной работы:
 - методика последовательности операций работы;
 - короткий текст в компьютере перед началом работы;
 - методические указания студентам о последовательности действий при выполнении работы;
 - методические указания оператору;
 - схема установки и функциональных элементов (приборов);
 - указаний пределов геометрии, частоты, вида объекта, пределов электрических параметров;
 - форма таблиц и графиков;
 - вид отчета, указания по анализу результатов;
 - выполнение всех экранов для ввода в «SYDNEY».
4. Подготовка математической модели и ее реализация:
 - Набор формул к сценам;
 - Оценка пределов изменений;
 - Маткад ли Раскаль модули;
 - Отработка модели на компьютере;
 - Графическое представление материалов;
 - Создание и анализ графиков;
 - Расчет одного типичного случая и поверка результатов компьютерных данных
5. Подготовка и выполнение фотографий лабораторной установки и её элементов
6. Озвучивание фотографий, сопровождающего объяснительного материала, сопряжение элементов сценария.
7. Ввод мультимедийных элементов: указатели, стрелки, сопровождающие надписи и т.д.
8. Окончательная проверка, введенных сценариев в систему «SYDNEY».
9. Организационная работа курирующего лабораторную работу

7. Изготовление тестового материала.

Построитель тестов

«Построитель тестов» – это инструментальная программа, входящая в состав интегрированной системы обучения «Sydney» и предназначенная для подготовки тестов, используемых в качестве тренажеров, задачников, контрольных и экзаменов.

Принцип работы «Построителя тестов»

Создание тестов с помощью данной программы происходит в два этапа. На первом этапе составляется текстовый документ в формате RTF (Rich Text Format) в подходящем текстовом редакторе, например, в Microsoft Word. Этот текстовый документ содержит тестовые вопросы и ответы на них, а также дополнительную информацию о тесте, например, название теста, имя автора и т.д. Формат RTF позволяет использовать в тексте шрифты различного написания, цвета и размера. Кроме этого, документ может содержать рисунки, формулы, таблицы и др. Такой подход к подготовке методического материала имеет несколько положительных сторон:

- преподавателю, составляющему тест, необходимо уметь работать только с текстовым редактором и не нужно владеть приемами работы с какой-нибудь инструментальной программой, а тем более уметь программировать;
- использование текстового редактора в качестве инструмента упрощает операции редактирования содержания теста, т.к. сводит эти операции к работе с фрагментами текста (вставка, удаление, копирование);
- форма представления методического материала, подготовленного в текстовом редакторе, является универсальной, и этот материал может быть легко преобразован в другую форму, например, его можно распечатать на принтере.

На втором этапе создания теста берется готовый текстовый документ и с помощью «Построителя тестов» конвертируется в файл теста. Для того, чтобы программа могла правильно выделить из текстового документа вопросы и ответы, на этапе подготовки документа в текст вставляются специальные команды языка разметки теста. Вставка этих команд в текст документа автоматизирована в «Построителе тестов». При использовании текстового редактора Microsoft Word автоматизация вставки команд разметки осуществляется с помощью макросов, специально для этого разработанных.

Команды языка разметки теста

Система команд «Построителя тестов» размечает весь документ на определенные фрагменты: текст вопроса, текст числового ответа, текст ответа многовариантного выбора и т.д., и поэтому является языком разметки документа. Команды разметки представляют собой несколько символов, заключенных в угловые скобки. Все команды в «Построителе тестов» имеют как английский вариант написания, так и русский (Таблица 1). Команды

закрываются в угловые скобки, например <about>. Эта же самая команда, но в русском варианте, будет выглядеть так: <название>.

Для того чтобы разметить какой-либо фрагмент текста, нужно указать, где этот фрагмент начинается, а где заканчивается. Для этого существуют открывающие команды и закрывающие команды. Открывающая команда начинает фрагмент, а закрывающая команда заканчивает этот фрагмент. Открывающая и закрывающая команды должны отличаться только символом косая черта «/», который стоит после угловой скобки закрывающей команды. Например, следующий фрагмент документа задает название теста: <about>Простой тест, написанный в качестве примера</about>.

В некоторых командах языка разметки теста можно указывать параметры. Параметр пишется между угловыми скобками открывающей команды, например, параметр **count**, означающий количество задач по определенной теме, указывается так: <topic count=1>Название темы</topic>. У одних параметров, таких как в этом примере, после знака равно «=» должно указываться значение параметра, а другие параметры указываются без значения, например, параметр **horizontal**, указывающий на горизонтальное расположение панели вопроса, записывается так: <q horizontal>Сколько будет дважды два?</q>. Возможные параметры команд и их описание приведено в справочной таблице «Команды языка разметки теста» (Таблица 1).

Структура теста

Обычно, тест состоит из нескольких тестовых задач, например, из ста, которые называют банком задач. Учащемуся для тестирования предлагают только часть задач, выбранных из банка задач случайным образом или по какому-нибудь критерию, например, в зависимости от их сложности. Количество задач, предъявляемых учащемуся, задается командой <count> (количество) следующим образом: <count>10</count>.

Все тестовые задачи в банке задач могут быть сгруппированы по каким-либо темам. Тогда необходимо указать количество задач, предъявляемых учащемуся по каждой теме. Для этого используется команда <topic> (тема) с указанием значения параметра **count**: <topic count=10>Название темы</topic>. Все задачи, текст которых идет после этой команды, будут принадлежать к этой теме, поэтому необязательно указывать тему перед каждой задачей. Чтобы начать новую тему, достаточно вставить в текст аналогичную команду, указав название новой темы.

Структура тестовой задачи

Тестовая задача состоит из вопроса и ответа. Кроме этого, тестовая задача может содержать пояснение к вопросу. Текст вопроса и пояснения представляются учащемуся в формате RTF, т.е. в вопросе и пояснении можно использовать шрифты различного размера, цвета и стиля, иллюстрации, формулы и др. объекты. Текст вопроса помечается командой <q> (сокращение английского слова question) или командой <вопрос>, например,

так: `<q>Сколько пальцев у гориллы?</q>`. Текст пояснения помечается командой `<d>` (сокращение английского слова description) или командой `<описание>`, например, так: `<d>А сколько у Вас?</d>`.

Типы ответов

Тестовые ответы бывают закрытого типа и открытого типа. К ответам закрытого типа относятся ответы множественного выбора, т.е. учащемуся необходимо выбрать один или несколько правильных ответов из множества предложенных ответов. К ответам открытого типа относятся ответы, которые учащийся должен ввести сам. Отвечая на задачи открытого типа, учащийся не ограничен каким-либо выбором. Для того, чтобы автоматически проверить правильность ответа учащегося, нужно задать эталон ответа.

Тесты в системе обучения Sydney могут содержать ответы следующих типов:

1. Открытого типа
 - 1.1. Выбор из строк
 - 1.2. Выбор из картинок
 - 1.3. Упорядочивание строк
2. Закрытого типа
 - 2.1. Числовой ответ
 - 2.2. Строковый ответ
 - 2.3. Свободное изложение (текст)
 - 2.4. Рисунок
 - 2.5. Текст и числовой ответ
 - 2.6. Текст и строковый ответ
 - 2.7. Ответ, проверяемый в «СИМВОЛ-ИДС»

Ответ множественного **выбора из строк** задается командой `<a>` (от английского слова answer) или командой `<ответы>`. Между открывающей и закрывающей командой помещаются все варианты ответа, причем каждый ответ должен начинаться с новой строки. Правильный ответ или правильные ответы, если их несколько, помечаются символом `@` в начале строки. Варианты ответов не должны быть слишком длинные, так как они предъявляются учащемуся по одному ответу в каждой строке, поэтому длинный ответ не влезет в экран, и, чтобы его прочесть, необходимо будет воспользоваться полосой прокрутки. Варианты ответов этого типа не должны включать в себя рисунки и формулы, т.к. учащемуся будет предъявлен только текст ответов. Варианты ответов должны быть написаны одним шрифтом, одним цветом, одним размером и стилем. Пример:

`<q>В чем измеряется напряжение?</q>`

`<a>в амперах`

`@в вольтах`

`в омах`

Если эти возможности не удовлетворяют потребностей, то можно воспользоваться ответом в виде выбора из картинок.

Ответ множественного **выбора из картинок** или формул задается командой `<i>` (от английского слова image) или командой `<картинки>`. Каждая отдельная картинка, которая будет находиться между открывающей и закрывающей командой, станет одним из вариантов ответа. Но текст, находящийся между этими картинками, не будет обрабатываться конвертером и не войдет в ответы. Правильный ответ, также как и в предыдущем случае, отмечается знаком `@`, стоящим перед картинкой. В следующем примере формулы набраны в редакторе формул:

`<q>Какой формулой описывается закон Ома?</q>`

`<i> @ $I = \frac{U}{R}$ $U = \frac{I}{R}$ $R = \frac{I}{U}$ @ $U = R \cdot I$ </i>`

В обоих типах множественного выбора правильный ответ может быть один или их может быть несколько. Если правильный ответ один, тогда тестирующая программа автоматически даст учащемуся возможность отмечать только один ответ. Есть возможность управлять этим процессом. Для этого у команд множественного выбора есть параметр **radio**. Если указать значение параметра `radio=off`, тогда тестирующая программа даст учащемуся возможность отметить сколько угодно ответов, даже если правильный ответ всего один.

Следующий тип ответа называется **упорядочивание строк**. Учащемуся предъявляется несколько строк текста, которые он должен выстроить в определенной последовательности путем перемещения этих строк. Строки задаются в необходимом порядке с помощью команды `<order>`, а перед показом учащемуся случайным образом перемешиваются. Строки должны отделяться друг от друга символом «возврата каретки», т.е. новую строку нужно начинать нажатием клавиши Enter. Вот пример тестовой задачи с этим типом ответа:

`<q>Расположите события в хронологическом порядке:</q>`

`<order>Постройка пирамиды Хеопса`

Олимпийские игры

Основание Рима

Рождество Христово</order>

Числовой ответ задается командой `<n>` (от английского слова number) или командой `<число>`. Числовой ответ относится к открытому типу ответов, т.е. учащемуся в качестве ответа будет необходимо ввести какое-либо число. Эталонный ответ может задаваться с некоторой точностью, для чего нужно после правильного ответа указать точность, разделив ответ и его точность пробелом, например, так: `<n>3.14 0.01</n>` (правильным ответом будет любое число из интервала 3.14 ± 0.01). Эталонный ответ может задаваться арифметическим выражением с использованием элементарных функций, например, `<n>A*sin(2*pi*f) 0.01</n>`. При этом параметры, используемые в выражении, задаются в секции констант - `<const>A=rnd*10; f=100e3+rnd*100e3</const>`. В приведенном примере эталон числового ответа является случайной величиной.

Строковый ответ, который также как и числовой ответ относится к открытому типу ответов, задается командой `<s>` (от английского слова string) или командой `<строки>`. В качестве ответа учащийся должен будет ввести несколько символов, например, какое-нибудь слово или словосочетание (например: `<s>Москва</s>`). В одной тестовой задаче может быть не более четырех строковых ответов. Ответ учащегося считается правильным, если он совпадает с эталонным ответом. При вводе ответов с клавиатуры нередко происходят случаи ввода ответа с ошибкой, поэтому при сравнении ответа учащегося с эталоном применяют корректировку ошибок ввода. Режим корректировки ошибок ввода и нечувствительности к регистру символов по умолчанию включен. В этом режиме маленькие и большие буквы не отличаются, и учащийся может ошибиться на один символ. Правильным ответом в приведенном примере может быть «москва», или даже «масква». Если регистр символов важен для ответа, то этот режим можно выключить, задав параметр `case=off`. Например, `<s case=off>AaBb</s>`. В некоторых случаях правильный ответ не может быть задан однозначно, тогда в качестве эталона ответа необходимо использовать все возможные правильные ответы, перечисленные через запятую и заключенные в фигурные скобки, например, так {моряк, матрос}.

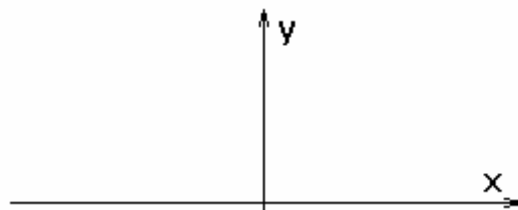
Два следующих типа ответа не обрабатываются автоматически. Ответы учащегося сохраняются в протоколе тестирования и могут быть проверены преподавателем с помощью специальной программы просмотра результатов тестирования. На предварительную оценку за тест эти ответы никак не влияют. Один из них является ответом **свободного изложения** и задается командой `<t>` (от английского слова text) или командой `<текст>`. В качестве ответа учащийся может написать любой текст, например, сочинение, решение задачи, определение и т.д. В качестве эталона ответа можно использовать предложение написать какой-нибудь текст, например, `<t>Здесь напишите определение ...</t>`.

Другой тип ответа является ответом в виде **рисунка**, который учащийся может нарисовать с помощью простого графического редактора, встроенного в тестирующую программу. Этот тип ответа задается командой `<p>` (от английского слова picture) или командой `<рисунок>`. В качестве рисунка, который должен присутствовать между открывающей и закрывающей командой, можно использовать чистую картинку или какой-нибудь шаблон рисунка, например координатные оси.

`<q>Нарисуйте график функции`

`<p> $\frac{\sin(x)}{x}$ </q>`

`</p>`



Параметры теста

Параметры теста определяют свойства теста. Изменяя некоторые параметры теста, можно получить разнообразные учебные программы, основанные на тесте. Обычно, параметры теста задаются в начале файла перед тестовыми задачами. К параметрам теста относятся:

- Название теста (команда `<about>` или `<название>`).
- Автор теста (команда `<author>` или `<автор>`).
- Уникальный индекс теста (команда `<index>` или `<индекс>`). Индекс необходим для корректной работы системы и указывается ее разработчиком.
- Время тестирования (команда `<time>` или `<время>`). Время, указанное в минутах, определяет длительность тестирования. При тестировании время, потраченное на тест, показывается учащемуся, и после окончания времени тестирования процесс тестирования автоматически заканчивается. Если нет необходимости ограничивать время тестирования, то указывается параметр **unlimited**, например, так: `<time unlimited>45</time>`.
- Оценка за тест (команда `<mark>` или `<оценка>`). Для оценивания ответов на тестовые задачи применена пропорциональная шкала оценивания. Чтобы задать оценочную шкалу, необходимо указать два числа – максимальную и минимальную оценку. Например, при использовании пятибалльной системы оценок указывается пять и ноль. Если тестовых задач будет 25, тогда пятерку учащийся получит при правильном ответе на 23-25 задач, четверку – 18-22, тройку – 13-17, двойку – 8-12, единицу – 3-7, ноль – 0-2. Кроме оценки очень часто требуется получить решение об успешности теста (зачет – незачет). Для этого необходимо указать третье число – пороговую оценку за тест. Тогда решение об успешности тестирования будет принято, если оценка за тест, полученная учащимся, больше или равна пороговой оценке. Пример: `<mark>5 0 3</mark>`.
- Тип теста (команда `<type>` или `<тип>`). Указание типа теста предъявляет определенные требования программе тестирования. Например, тип **trainer** указывает на то, что тест предназначен для использования в качестве тренажера, т.е. ответы учащегося будут оцениваться сразу после ввода, а временные и другие ограничения будут отсутствовать. А тип **topic** указывает на то, что выборка из банка задач этого теста будет производиться только по определенной теме, в отличие от типа **examine**, в котором выборка задач производится по всем темам теста. Тип теста по умолчанию – **examine**.

Пример теста

```
<about>Простой тест</about>
```

```
<author>Иванов Иван Петрович</author>
```

```
<index>1000</index>
```

```
<mark>5 1 3</mark>
```

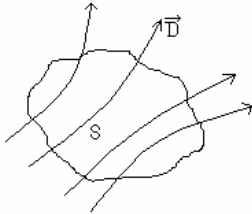
```
<time>10</time>
```

```
<topic count=1>Пример по электродинамике</topic>
```

```
<q>Как называются среды, в которых один из параметров  $\varepsilon$ ,  $\mu$  и  $\delta$  (или два, или все) зависят от координат?</q>
```

<a>линейные
@неоднородные
изотропные
анизотропные
однородные

<q>Чему равен поток вектора \mathbf{D} через поверхность S , изображенную на рисунке?



</q>

<i> $\oint_S \bar{D} d\bar{S} = q$ $\oint_S \bar{D} d\bar{S} = -q$ $\oint_S \bar{D} d\bar{S} = \int_V \rho dV$ $\oint_S \bar{D} d\bar{S} = \int_V \frac{\partial B}{\partial t} dV$ @ $\oint_S \bar{D} d\bar{S} = 0$ </i>

<topic count=1>Пример по генетике</topic>

<q>Синдактилия (сросшиеся пальцы) наследуются как доминантный аутосомный признак. Один из родителей гетерозиготен, другой гомозиготен по рецессивному аллелю. Какова вероятность рождения здорового ребенка в этой семье?</q>

<d>Ответ: 1/2</d>

<s>1/2</s>

<topic count=1>Пример по программированию</topic>

<q>Вычислите площадь фигуры ограниченной кривой $\frac{1}{7 * \cos^2 x + 16 * \sin^2 x}$, прямыми $x=0$, $x=\pi$ и осью Ox .

Примечание: вычисление интеграла производить методом прямоугольников.</q>

<d>Для вычисления площади нужно вычислить интеграл $\int_0^\pi \frac{1}{7 * \cos^2 x + 16 * \sin^2 x} dx$
Вычисление интеграла произведем методом прямоугольников. Промежуток интегрирования (a, b) делим точками x_1, x_2, \dots, x_{n-1} на n равных частей $x_0 = a$, $x_n = b$

длина каждой $h = \frac{b-a}{n}$, полагая, что $f(x_0) = y_0, f(x_1) = y_1, \dots$ и т.д., тогда

$$\int_a^b f(x) dx \approx \frac{b-a}{n} [y_0 + y_1 + \dots + y_{n-1}]$$

</d>

<n>0.297 0.001</n>

<t>Здесь напишите программу, с помощью которой решается данная задача</t>

<end>

Таблица 1

Команды языка разметки теста

Команда	Параметры	Описание команды
---------	-----------	------------------

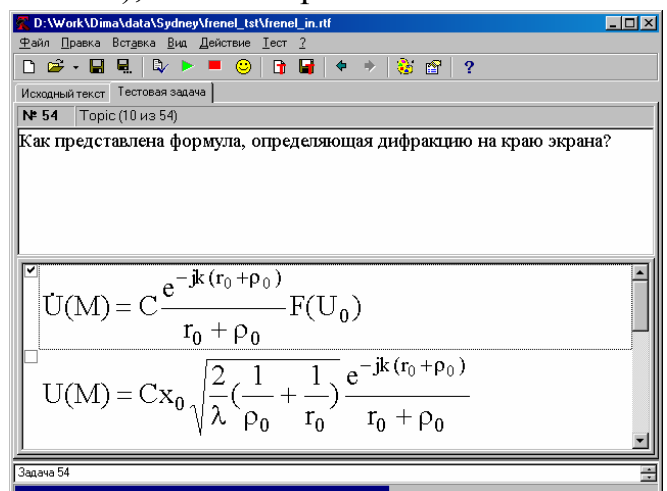
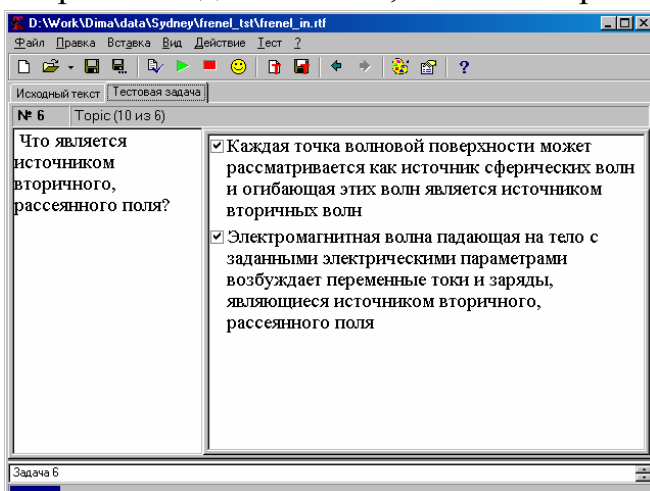
<about>, <название>		Описание теста, название.
<author>, <автор>		Автор теста
<index>, <индекс>		Уникальный индекс теста.
<time>, <время>	unlimited	Время выполнения теста в минутах. Если присутствует параметр unlimited , то время не ограничивается.
<mark>, <оценка>	hide	После этой команды пишутся три числа – максимальная оценка, минимальная оценка, пороговая оценка (при получении оценки выше пороговой тест считается сданным успешно). Числа разделяются пробелом. Параметр hide скрывает информацию о результате тестирования от учащегося.
<type>, <тип>	description = on off after	Тип теста. Может принимать значения: topic - для предъявления учащемуся выбираются задачи по одной из нескольких тем, examine - для предъявления учащемуся задачи выбираются по всем темам теста, trainer – тест предъявляется учащемуся в качестве тестового тренажера. Параметр description задает режим показа описания вопроса <d> и может принимать значения: off (не показывать описание), after (показывать после теста для задач, решенных неправильно) и on (показывать описание во время тестирования).
<instruction>, <инструкция>		Инструкция по выполнению тестовых заданий (появляется перед началом тестирования).
<count>, <количество>		Количество задач, выбираемых случайным образом из всех задач данной темы, для предъявления учащемуся.
<topic>, <тема>	count = n	Название темы. В тесте может быть несколько тем. Если данная команда отсутствует, то тема одна. Параметр count задает количество задач по данной теме, выбираемых случайным образом для представления учащемуся, например, count=5. Работает аналогично команде <count>.
<q>, <вопрос>	width = n %, vertical horizontal	Тестовый вопрос. Может содержать рисунки и формулы. Параметр width задает ширину панели вопроса в процентах или пикселах (width=85% или width=250). Параметры vertical и horizontal задают вертикальное или горизонтальное расположение панелей вопроса и ответа.

<d>, <описание>	exec	Правильный ответ, пример решения, наводящий вопрос и т.д. Может содержать рисунки и формулы. Если "описание" отсутствует, то оно автоматически генерируется в виде правильного ответа. "Описание" в зависимости от параметров теста может быть доступно учащемуся во время тестирования или после тестирования, либо доступ к "описанию" может отсутствовать. Параметр exec означает, что описание является какой-либо обучающей программой. В этом случае необходимо указать уникальный индекс обучающего модуля.
<const>, <константы>	digits = n	Константы, которые вычисляются и могут быть использованы в тексте вопроса и пояснения, а также при вычислении эталона числового ответа. Параметр digits задает количество цифр, отображаемое при выводе значения константы. По умолчанию digits=4
<a>, <ответы>	radio = on off	Выбор из строковых ответов. Каждый строковый ответ должен начинаться с новой строки и не может содержать рисунки и формулы. Текст ответа не может иметь различное форматирование, например нельзя использовать различные шрифты, цвета, размер. Правильный ответ отмечается символом @ в начале строки. Параметр radio задает возможность множественного или единичного выбора ответа (radio=off или radio=on). По умолчанию устанавливается единичный выбор, если правильный ответ один.
<n>, <число>		Числовой ответ. Через пробел после числового ответа записывается точность ответа.
<s>, <строки>	case = on off msprint=on off	Строковый ответ (слово). Может быть не более четырех строковых ответов, каждый ответ должен начинаться с новой строки. Параметр case задает чувствительность к регистру строкового ответа при проверке (case=on или case=off). По умолчанию чувствительность к регистру отсутствует. Параметр misprint задает чувствительность к ошибкам ввода, т.е при включенной чувствительности к ошибкам ввода допускается ввод строкового ответа с ошибкой в один символ. По умолчанию чувствительность к ошибкам ввода включена. Возможно задавать несколько эталонов ответов с помощью

		фигурных скобок, например {моряк, матрос}
<i>, <картинки>	radio = on off	Выбор из картинок или формул. Текст, находящийся между картинками, будет игнорироваться. Правильный ответ отмечается символом @ перед картинкой. Параметр radio задает возможность множественного или единичного выбора ответа (radio=off или radio=on). По умолчанию устанавливается единичный выбор, если правильный ответ один.
<t>, <текст>		Текстовый ответ свободного изложения. Здесь может содержаться шаблон ответа, приглашение написать ответ и т.д. Этот ответ автоматически не проверяется и не учитывается при выставлении оценки.
<p>, <рисунок>		Ответ в виде рисунка (свободное изложение). Может быть чистым или содержать вспомогательный рисунок, например, координатные оси. Этот ответ автоматически не проверяется и не учитывается при выставлении оценки.
<sim>, <СИМВОЛ>		Код задачи в системе "Символ-ИДС"
<order>		Упорядоченные строки. Перед предъявлением учащемуся порядок строк нарушается. Учащемуся требуется его восстановить.
<end>, <конец>		Конец теста

7.1 Программа TBuilder.

Эта программа предназначена для компиляции тестов. Тесты должны быть написаны по принципу гипертекстовой разметки. Это значит, что каждые области, такие как заголовок, вопрос, ответ, должны быть ограничены тэгами. Тэг – это символ/текст ограниченный кавычками < >. Например, область вопроса начинается с открывающего тэга <q> и заканчивается закрывающим тэгом </q>. Такое построение исходного текста тестового материала позволяет разграничить смысловые области и добавить необходимые данные для форматирования вопросов. Например, если тэг <q> форматирует тест так, что вопрос находится слева, а ответ справа (рис 2.1а), то тэг <q horizontal> ставит



вопрос на верх, а ответ вниз (рис 2.1б).

а)

б)

Рисунок 7.1 Возможные расположения ответов.

Любой тест должен начинаться с заголовка. Заголовок содержит информацию о тесте и авторе и должен состоять из следующих блоков:

<about> ... </about> - тема / название теста,

<author> ... </author> - автор теста,

<time> ... </time> - время, отведенное на прохождение теста,

<index> ... </index> - уникальный индекс, выделяемый администратором Sydney'я для каждого методического ресурса,

<count> ... </count> - число вопросов, выбираемые из всего множества вопросов данного теста,

<mark> ... </mark> - оценки, получаемые за тест: максимальная, минимальная и проходная.

Заголовок также может содержать необязательные блоки:

<instruction> ... </instruction> - инструкции по тесту, отображаются в информационном окне перед прохождением теста.

<type> ... </type> - тип теста: traner – режим тренажера; examine – экзаменационный режим.

Пример заголовка:

```
<about> Входной контроль к ЛР "Исследование зон Френеля"</about>
```

```
<author> Шангина Л.И. </author>
```

```
<mark hide>5 1 4</mark>
```

```
<time>30</time>
```

```
<index>4135</index>
```

```
<count>10 </count>
```

```
<instruction>Для допуска к лабораторной работе студент должен получить положительную оценку, ответив на предлагаемые вопросы, которые выкидываются вероятностным образом из банка входных контролирующих вопросов.</instruction>
```

Следует учитывать, что теги могут меняться от одной версии программы к другой, поэтому точное их описание нужно смотреть в пункте меню программы “? | Команды”.

После заголовка идет тело теста. Вопрос выделяется тегами <q> и </q>, текстовые ответы на вопрос тегами <a> и , графические ответы тегами <i> и </i>, числовые ответы тегами <n> и </n>. У вопроса может быть только один тип ответов. Каждый из ответов пишется на отдельной строке. Перед правильными ответами ставится символ @.

В конце теста ставится завершающий тег <end>. Исходный текст теста должен сохраняться формате RTF.

Перед компиляцией теста его надо проверить на наличие ошибок. Компилятор проверяет текст теста на правильность последовательности тегов.

При обнаружении ошибки, программа выведет в строке состояния, находящейся внизу окна, информацию об ошибке и номере сбойного вопроса теста.



Рисунок 7.2 Панель инструментов.

Компиляция может проводиться в двух режимах: пошаговом и автоматическом. При пошаговом режиме компиляции необходимо для каждого вопроса нажимать кнопку конвертирования (F9). При этом текущий вопрос отображается в окне "Тестовая задача". В строке состояния отображается информация о откомпилированных вопросах и найденных ошибках. Переход в автоматический режим производится нажатием на кнопку "Автоматическое конвертирование". Остановить конвертирование можно нажатием кнопки "Остановить действие".

Программа предоставляет возможность создания нового и редактирования уже существующего теста. При этом вставка тегов может производиться путем ручного набора с клавиатуры либо через пункт меню "Вставка".

7.2 Программа TCheck.

После компиляции исходного текста теста программой Tbuilder получается готовый тест, который можно уже вставлять в систему. Однако, бывают случаи, когда в тесте попадают ошибки, связанные с неправильным форматированием текста или с опечатками наборщика теста. Все такие ошибки необходимо выявлять до встраивания теста в Sydney. Программа Tcheck предоставляет возможность просмотра теста с произвольным доступом к вопросам. При этом вопросы и ответы отображаются в том виде, в котором тест будет просматриваться пользователем.

При входе в программу отображается окно, предназначенное для выбора директории, в которой находятся тесты (рис. 7.3). Если в выбранной директории есть тесты, то они отображаются в правой части окна. Для запуска проверки нужно выделить тест и нажать на кнопку "Пуск" программы.

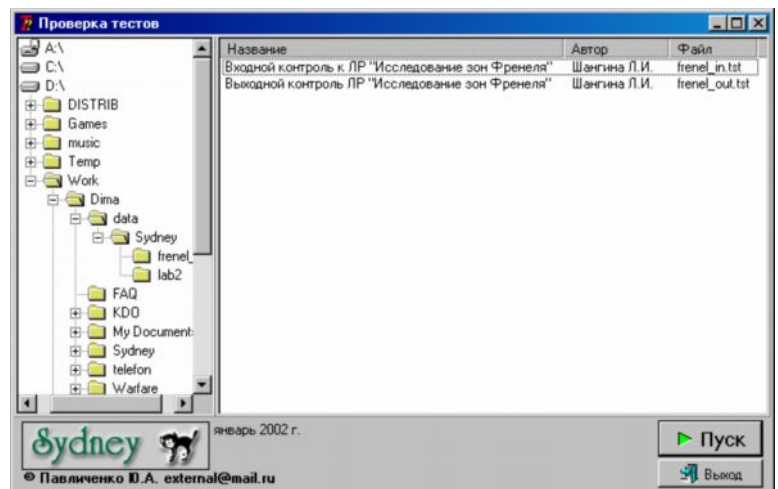


Рисунок 7.3 Окно выбора теста

При входе в программу отображается окно, предназначенное для выбора директории, в которой находятся тесты (рис. 7.3).

Если в выбранной директории есть тесты, то они отображаются в правой части окна. Для запуска проверки нужно выделить тест и нажать на кнопку “Пуск” программы.

Вид окна программы при проверке теста представлен на рисунке 7.4.

При проверке вопроса нужно выбрать один из вариантов ответа и нажать на кнопку “Проверка”. Если ответ правильный, то в верхней части окна программы высветится сообщение “Правильно”. В противном случае высветится “Неправильно”, но при условии, что на вопрос есть только один ответ. Если ответов несколько, и выбраны не все ответы, высветится сообщение “Частично”. Такое же сообщение высветится, если среди правильных ответов будут выбраны неправильные. Ведется протокол результатов проверки вопросов теста.

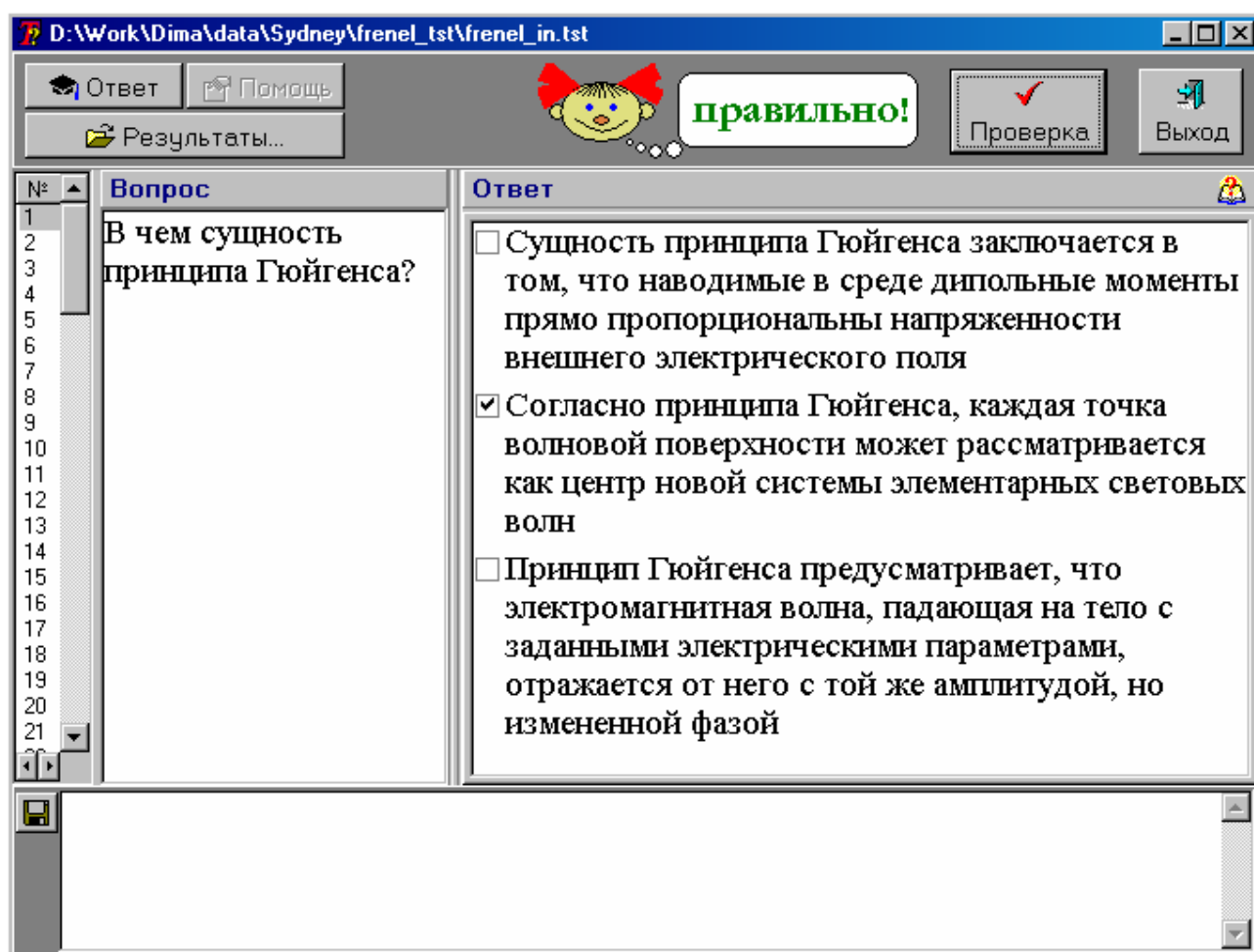


Рисунок 7.4 Окно проверки теста.

Основные концепции системы SYDNEY обсуждались на нескольких международных, всероссийских и региональных конференциях. С января 2000 г. идет эксплуатация программного обеспечения локальных центров с готовыми курсами Windows, Word, Excel по обучению компьютерной грамотности населения объемом от 40 до 70 часов в Кемеровской, Новосибирской, г. Северске Томской областях.

Литература

1. Павличенко Ю.А., Хатьков Н.Д. Методические ресурсы преподавателя в мультимедийной среде Sydney и временные затраты на его создание. 2-я Всероссийская конференция «Электронные учебники и электронные библиотеки в открытом образовании», с. 322 – 329, МЭСИ, 29 ноября 2001г., г. Москва.
2. Майоров А.Н. Теория и практика создания тестов для системы самообразования. Изд-во: "Народное образование", 2000г, с. 352.
3. Павличенко Ю.А., Хатьков Н.Д., Шангина Л.И. Подготовка электронных экзаменуемых материалов в системе мультимедийного обучения Sydney. Труды IV Международной научно-практической конференции «Высшее техническое образование: качество и интернационализация» Томск, 14-17 марта 2000г. – с.48.
4. Павличенко Ю.А., Хатьков Н.Д. Подготовка тестов с помощью языка разметки. Тезисы докладов региональной научно-методической конференции, Томск, 1-2 февраля 2001г., с.125-127
5. Павличенко Ю.А. Хатьков Н.Д. Создание лабораторных работ в мультимедийной системе компьютеризированного обучения “Sydney”. XII международная конференция выставка «Информационные технологии в образовании»: Сборник трудов. Часть IV. – М.:МИФИ, 4-8 ноября 2002 – 92-93с.
6. Павличенко Ю.А., Хатьков Н.Д., Шангина Л.И. Оценка эффективности электронных средств дистанционного обучения. Научно-методическая конференция «Современное образование: системы и практика обеспечения качества», 29-30 января 2002г., г.Томск, с.107-108
7. Интегрированная мультимедийная система дистанционного обучения Sydney, рег. №990548 от 29 июля 1999г., РОСПАТЕНТ