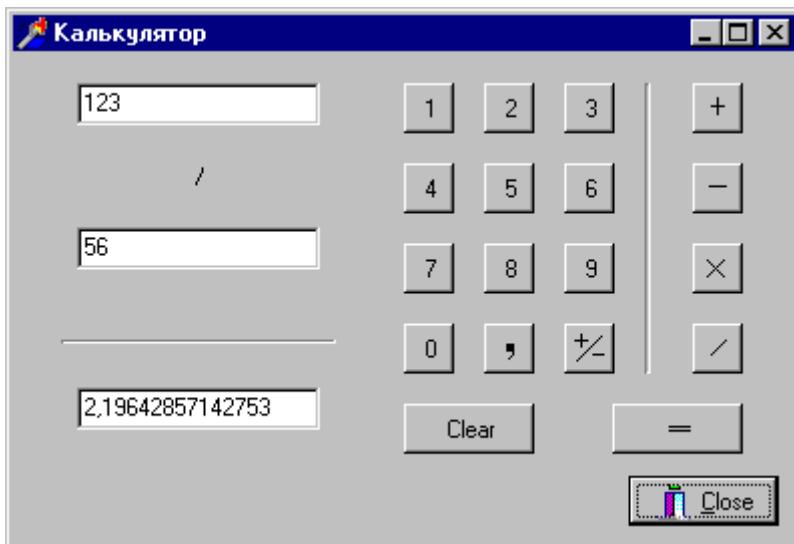


Лабораторные работы

Задание 1. Калькулятор

Задача: Создать приложение, выполняющее функции простейшего калькулятора (сложение, вычитание, деление, умножение). Предусмотреть защиту от повторного нажатия на кнопку десятичной запятой и выполнения операции деления на ноль. При повторном нажатии на кнопку знака числа знак вводимого числа должен меняться на противоположный. Кнопка "Clear" очищает все поля ввода. Знак выполняемой операции отражается в окне приложения между полями ввода операндов.



Компоненты:

- StaticText – окна для ввода чисел и вывода результата;
- Label – выполняемая арифметическая операция;
- Button – кнопки цифр, операций, очистки (Clear) и получения результата (=);
- BitBtn – кнопка закрытия приложения (Close);

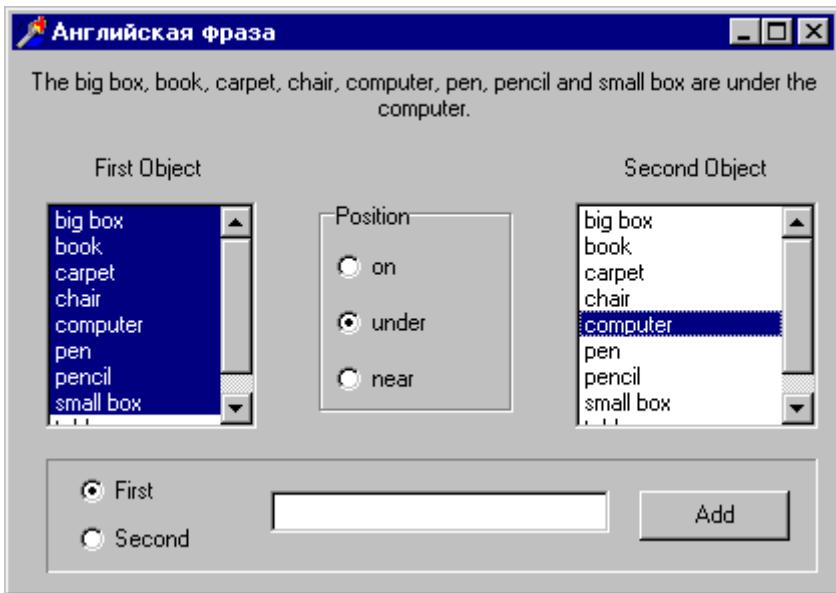
- Bevel – разделитель для кнопок цифр и кнопок операций,
разделитель для operandов и результата.

Задание 2. Генератор английских фраз

Задача: Создать приложение для генерации английской фразы.

После запуска программы появляется предложение, состоящее из слов, стоящих первыми в каждом списке (слова должны быть выделены). После выбора в первом списке одного слова или группы из нескольких необязательно подряд идущих слов предложение должно мгновенно измениться. Аналогично, после выбора одного слова из радиогруппы или одного слова из второго списка, предложение должно мгновенно измениться. При этом должны соблюдаться правила построения английских предложений: глагол *is* – в случае единственного подлежащего, глагол *are* – во множественном числе, а также расстановка запятых при перечислении дополнений и союза *and* перед последним словом в перечислении. Если предложение не помещается на одной строке, оно должно продолжаться на следующей.

Пользователь должен иметь возможность добавить новое слово в первый или во второй список. При нажатии на кнопку "Add" слово, введенное в окне ввода, добавляется в тот список, который выбран в радиогруппе слева от окна ввода. При этом должна проводиться проверка на наличие введенного слова в списке, и если это слово уже содержится в списке, то оно не добавляется.



Компоненты:

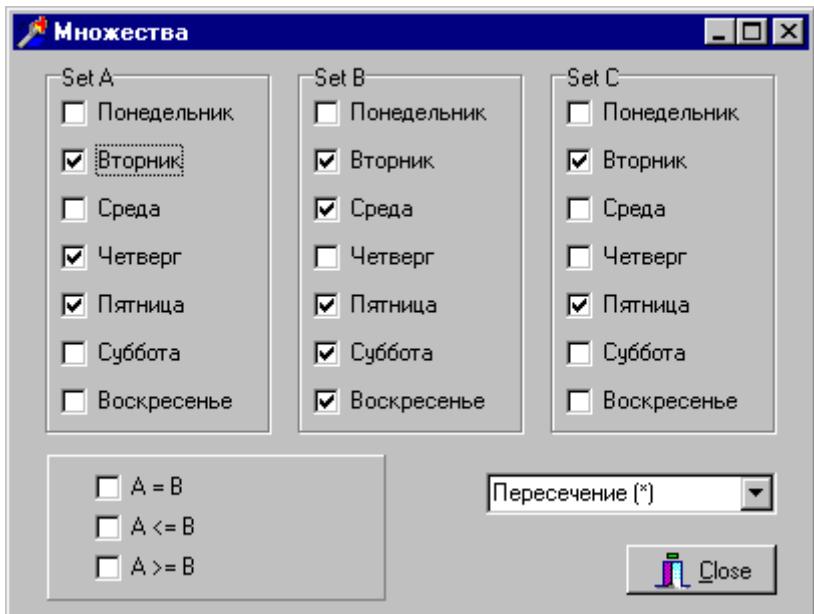
- Label – сгенерированная фраза, названия списков;
- RadioGroup – первая группа содержит список предлогов, вторая – номера списков;
- ListBox – содержит список подлежащих и список дополнений, в первом списке разрешен множественный выбор;
- Edit – окно ввода нового слова в один из списков в зависимости от выбора соответствующей радиокнопки слева от окна ввода;
- RadioButton – радиокнопка выбора списка: First – список подлежащих, Second – список дополнений;
- Button – кнопка добавления слова в один из списков;
- Bevel – рамка, объединяющая поле ввода, две радиокнопки и кнопку "Add".

Задание 3. Множества

Задача: Создать приложение, выполняющее логические операции над множествами. Пользователь щелчком мыши включает или исключает элемент одного из множеств SetA или SetB, результат одной из операций над множествами (объединение, пересечение, разность) мгновенно отражается в множестве SetC. Одновременно обновляется информация об отношениях между множествами SetA и SetB: A=B – множества совпадают, A<=B – первое множество содержится во втором, A>=B - второе множество содержится в первом. Выбор операции над множествами осуществляется из раскрывающегося списка.

Компоненты:

- | | |
|----------|---|
| GroupBox | <ul style="list-style-type: none">– содержит элементы каждого из трех множеств; |
| CheckBox | <ul style="list-style-type: none">– элемент множества (каждый компонент принадлежит одному из компонентов GroupBox), а также индикатор отношения между множествами (все индикаторы принадлежат панели Panel); |
| Panel | <ul style="list-style-type: none">– содержит три индикатора отношений между множествами; |
| ComboBox | <ul style="list-style-type: none">– раскрывающийся список для выбора операции над множествами, содержит три операции: объединение, пересечение, разность; |
| BitBtn | <ul style="list-style-type: none">– кнопка завершения работы программы (Close). |



Замечание: Приложение должно содержать единственную процедуру – отклик на событие "щелчок мыши", которая должна быть назначена всем компонентам CheckBox из первого и второго множеств. В этой процедуре должна обновляться информация о составе третьего множества и информация об отношениях между первым и вторым множествами. Для ссылки на элементы множеств следует воспользоваться свойством Controls компонентов GroupBox, которым принадлежат компоненты CheckBox. Изменение состава третьего множества и состояния индикаторов отношения между множествами должно осуществляться только программно, щелчок мыши по этим элементам не должен иметь обычного эффекта – выделение или снятие выделения.

Задание 4. Секундомер

Задача: Создать приложение, выполняющее функции секундометра (включить, остановить, очистить показания).



Компоненты:

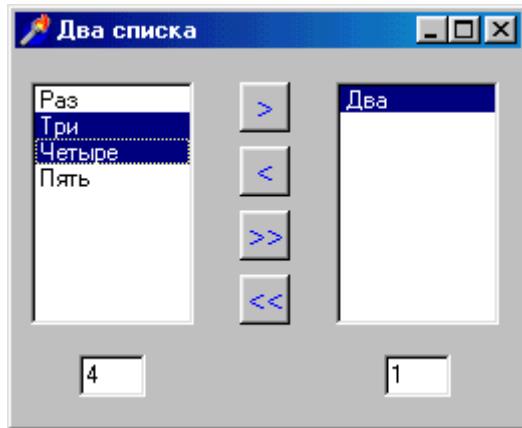
- Label – окна для вывода показаний времени (часы, минуты, секунды);
- Button – кнопки включения секундометра, остановки и очистки показаний;
- Timer – для отсчета времени.

Задание 5. Два списка

Задача: Создать приложение, выполняющее функцию перемещения строк между двумя списками. Два списка содержат несколько различных строк. Оба списка допускают множественный выбор. При нажатии на кнопку ">" выделенная строка или несколько выделенных строк переносятся из левого списка в правый. Это означает, что они удаляются из левого списка и добавляются в правый список. Аналогично работает кнопка "<", при нажатии на которую выделенные строки из правого списка переносятся в левый. При нажатии на кнопку ">>" все строки левого списка переносятся в правый список. При нажатии на кнопку "<<" все строки правого списка переносятся в левый список. В двух окошках под списками отражается текущее количество записей в каждом списке. При переносах эти значения должны мгновенно обновляться.

Компоненты:

- | | |
|---------|---|
| ListBox | – окна со списками; |
| Button | – кнопки ">", "<", ">>", "<<"; |
| Edit | – окна с количеством элементов в каждом списке. |



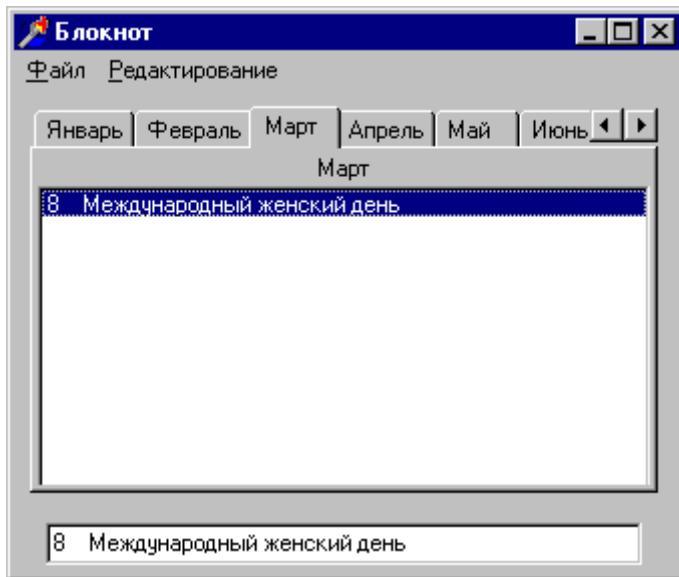
Задание 6. Блокнот

Задача: Создать приложение, выполняющее функции записной книжки с вкладками для каждого месяца одного года. Окно приложения содержит блокнот, на каждой странице которого находится список заметок для одного месяца. Окно ввода предназначено для отображения текущей выделенной строки в списке и для ввода новой строки. Меню имеет два пункта: "Файл" – включает пункты "Загрузить", "Сохранить" и "Выход" – для сохранения и загрузки всех записей блокнота в файле, и "Редактирование" – включает пункты "Добавить", "Изменить", "Удалить" – для работы с записями на одной странице блокнота.

Компоненты:

- | | |
|---------|---------------------------------------|
| Label | – название месяца на каждой странице; |
| ListBox | – список заметок на каждой странице; |

- PageControl – страницы блокнота;
MainMenu – основное меню программы;
Edit – окно отображения выделенной записи в блокноте, а также используется для ввода новой записи;
OpenDialog, SaveDialog – сохранение и загрузка файла с листами блокнота.



Задание 7. Телефонная книжка

Задача: Создать приложение для ведения телефонной книжки.

Таблица состоит из трех столбцов: фамилия, адрес, телефон, строки добавляются по мере ввода новых записей. Меню состоит из трех пунктов: "Файл" – включает пункты "Загрузить", "Сохранить", "Выход", "Редактирование" – включает пункты "Добавить", "Удалить",

"Отсортировать" и третий пункт "Поиск". При выборе пункта "Добавить" вызывается вторая форма с полями ввода для каждого столбца таблицы. Внешний вид формы представлен на рисунке. При нажатии кнопки "OK" в таблицу добавляется новая строка с введенными во второй форме значениями. При выборе пункта "Удалить" удаляется выделенная запись из таблицы, остальные строки поднимаются вверх. При выборе пункта "Отсортировать" строки таблицы сортируются по фамилии в лексико-графическом порядке. При выборе пункта "Поиск" вызывается вторая форма с одним доступным полем для ввода фамилии. После нажатия кнопки "OK" в таблице выделяется строка, соответствующая введенной фамилии, а если такой фамилии в таблице нет, то выдается сообщение "Объект не найден!".

The screenshot shows a Windows-style application window titled "Телефоны". The menu bar includes "Файл", "Редактирование", and "Поиск". The main area is a table with three columns: "Фамилия", "Адрес", and "Телефон". The data in the table is:

Фамилия	Адрес	Телефон
Иванов	ул.Лыжная, 11-7	77-55-66
Лебедев	пр.Ленина, 12-4	72-85-61
Петров	ул.Горького, 5-2	51-68-56
Рябов	пр.Калинина, 3-23	71-35-21

The screenshot shows a dialog box titled "Ввод значений". It contains three text input fields labeled "Фамилия", "Адрес", and "Телефон". Below the fields are two buttons: "OK" with a green checkmark icon and "Cancel" with a red X icon.

Компоненты:

- | | |
|-------------|---|
| StringGrid | – таблица, состоящая из трех столбцов, строки добавляются по мере ввода новых записей, одна строка зафиксирована; |
| MainMenu | – основное меню для первой формы; |
| Edit | – окна для ввода фамилии и адреса; |
| MaskEdit | – окно для форматированного ввода номера телефона; |
| BitBtn | – кнопки "OK" и "Cancel" на второй форме; |
| OpenDialog, | |
| SaveDialog | – сохранение и загрузка файла телефонной книги. |

Задание 8. Блоксхемы

Задача: Создать приложение для рисования диаграмм. Диаграмма состоит из прямоугольников, эллипсов и ромбов с текстом внутри, соединенных стрелками.

Всю область окна приложения занимает блокнот с тремя листами: "Фигуры", "Линии", "Рисунок".

На первом листе – "Фигуры" – определяются характеристики каждой фигуры диаграммы: размеры (высота и ширина), координаты верхнего левого угла (x, y), тип фигуры (прямоугольник, эллипс или ромб), цвета (контура фигуры, внутренней заливки, символов надписи).

Информация о фигуре хранится в объектах класса TMyFigure, связанных со списком компонента ListBox. Класс TMyFigure должен иметь поля, описывающие характеристики фигуры, конструктор и процедуру для рисования этой фигуры на произвольной канве.

При нажатии на кнопку "Добавить" к списку фигур добавляется новая строка с именем по умолчанию и с ней связывается новый объект класса TMyFigure. Все характеристики фигуры получают значения по умолчанию. При изменении хотя бы одной характеристики фигуры новое значение сразу же запоминается в соответствующем свойстве объекта.

При выделении какой-либо фигуры в списке значения ее характеристик должны мгновенно появиться в соответствующих полях

ввода, а при изменении хотя бы одной – сразу запоминаться, без выполнения дополнительных действий со стороны пользователя.

Изменение названия фигуры должно мгновенно отражаться в списке всех фигур. Тип фигуры выбирается только из списка.

Для полей ввода должны отслеживаться ввод неправильного значения и выход за диапазон значений.

При нажатии на кнопку "Удалить" из списка фигур удаляется выделенная строка и уничтожается соответствующий ей объект.

Щелчок мыши по прямоугольнику, обозначающему цвет, вызывает диалог выбора цвета, после закрытия которого прямоугольник окрашивается в выбранный цвет и цвет запоминается в соответствующем свойстве объекта выделенной в данный момент фигуры.

На втором листе – "Линии" – определяются стрелки, связывающие две фигуры. Для каждой стрелки определяется фигура и сторона фигуры, от которой начинается стрелка, и фигура и сторона фигуры, на которой заканчивается стрелка, и цвет линии.

Информация о линиях хранится в объектах класса TMyLine, связанных с таблицей StringGrid. Класс TMyLine должен иметь поля, ссылающиеся на фигуры, которые соединяются линией, и соответствующие стороны, а также цвет линии, конструктор и процедуру для рисования стрелки на произвольной канве.

Выбор фигуры выполняется только из списка с помощью кнопок ">". Список формируется автоматически и полностью соответствует списку на листе "Фигуры".

При нажатии на кнопку ">" выделенное название фигуры переносится в соответствующее поле ввода (начало линии или конец линии). Сторона присоединения линии выбирается из раскрывающегося списка. Щелчок мыши по панели "цвет" вызывает диалог выбора цвета, после закрытия которого внутренняя часть панели окрашивается в выбранный цвет. Этот цвет используется для рисования стрелки.

При нажатии на кнопку "Создать линию" в таблице добавляется еще одна строка, которая заполняется установленными в полях ввода значениями, и создается новый объект класса TMyLine, который связывается с первой ячейкой новой строки. Свойствам нового объекта присваиваются значения из полей ввода. Следует обязательно

предусмотреть обработку исключительной ситуации, когда отсутствует описание одного или обоих концов линии или концы совпадают.

При выборе какой-либо строки таблицы в соответствующих полях ввода мгновенно должны появиться значения характеристик выбранной линии.

При нажатии на кнопку "Удалить" из таблицы исключается выделенная строка и уничтожается объект, связанный с первой ячейкой этой строки.

На третьем листе – "Рисунок" – изображается диаграмма.

Главное меню формы состоит из двух элементов: "Файл" и "Нарисовать". Пункт "Файл" содержит следующие пункты: "Сохранить", "Загрузить", "Выход". Команда "Сохранить" сохраняет в выбранном пользователем файле информацию о всех фигурах и соединяющих их линиях, формат файла произвольный. Команда "Загрузить" заполняет список фигур и таблицу линий, команда "Нарисовать" перерисовывает диаграмму на третьем листе.

Компоненты:

- MainMenu – главное меню окна;
- ColorDialog – диалог выбора цвета;
- OpenDialog – диалог открытия файла;
- SaveDialog – диалог сохранения в файле;
- TabControl – блокнот с листами;
- Button – кнопки "Создать", "Удалить", "Добавить", "<", ">";
- Edit – поля ввода размеров, имени;
- Label – надписи на форме;
- Image – прямоугольники с цветом и диаграммой;
- StringGrid – таблица с описанием стрелок;
- ListBox – список фигур;
- ComboBox – раскрывающийся список для типа фигуры (содержит строки: прямоугольник, эллипс, ромб), раскрывающийся список для места присоединения стрелки (содержит строки: левая, правая, нижняя, верхняя);

- GroupBox – объединяет элементы управления на листе "Линии";
UpDown – используется в полях ввода.

