***Лабораторная работа №5. Управление калькулятором р-ичных чисел***

Тема: Классы Object Pascal, С++

*Цель: Сформировать практические навыки реализации классов средствами объектно-ориентированного программирования Object Pascal, С++.*

Задание

1. Разработать и реализовать класс “Управление калькулятором р-ичных чисел” тип TCtrl, используя класс

* Object Pascal,
* С++.

На Унифицированном языке моделирования UML (Unified Modeling Language) наш класс можно описатьть следующим образом:

|  |
| --- |
| **УправлениеКалькуляторомР-ичныхЧисел (тип TCtrl)** |
| состояниеКалькулятора: TCtrlState |
| редактор: TEditor |
| процессор: TProc |
| память: TMemory |
| число: TPNumber |
| выполнитьКомандуКалькулятора(a: Integer; var b, MState: String): String |
| выполнитьКомандуРедактора(a: Integer): String |
| выполнитьОперацию(a: Integer): String |
| выполнитьФункцию(a: Integer): String |
| вычислитьВыражение(a: Integer): String |
| установитьНачальноеСостояниеКалькулятора(a: Integer): String |
| выполнитьКомандуюПамяти(a: Integer; var MState: String): String |
| очиститьПамять(var MState: String): String |
| записатьВПамять(var MState: String): String |
| считатьИзПамяти(var MState: String): String |
| добавитьКЧислуВПамяти(var MState: String): String |
| читатьПисатьСостояниеКалькулятора: TCtrlState (свойство, опирающееся на поле) |
| выполнитьКомандуБуфераОбмена(a: Integer; var b: String): String |
| записатьВБуферОбмена(a: Integer; var b: String): String |
| читатьИзБуферОбмена(a: Integer; var b: String): String |
| сменитьОснованиеСистемыСчисления(b: integer; var s:String) |
| конструктор |
| деструктор |
| Обязанность:  управление выполнением команд калькулятора |

1. Класс должен отвечать за управление выполнением команд калькулятора. Класс должен обеспечивать:

* управление вводом и редактированием р-ичных чисел;
* управление выполнением операций, функций и выражений над р-ичными числами;
* управление выполнением операций по работе с памятью, объёмом в одно значение типа р-ичное число;
* управление работой с системным буфером обмена.

1. Протестировать каждый метод класса и класс в целом.

Рекомендации к выполнению

1. Класс TCtrl реализуйте в отдельном модуле UControl.
2. В модуле опишите перечисляемый тип TCtrlState = (cStart, cEditing, FunDone, cValDone, cExpDone, cOpChange, cError) для обозначения состояний калькулятора: cStart (Начальное), cEditing (Ввод и редактирование), cExpDone (Выражение вычислено), cOpDone (Операция выполнена), cValDone (Значение введено), cOpChange (Операция изменена),. cError (Ошибка).
3. В классе опишите следующие атрибуты:

* “редактор” - объект типа TEditior;
* “процессор” - объект типа TProc (TCalc в предыдущей редакции);
* “память” - объект типа TMemory;
* “состояние калькулятора” - тип TCtrlState;
* “число” - объект типа TPNumber (результат выполнения последней команды).

1. Набор основных операций класса определяется набором команд калькулятора, заданных пользователем. Кроме того, в классе будут определены вспомогательные операции, обеспечивающие реализацию основных операций. В классе опишите следующие операции:

* “выполнитьКомандуКалькулятора” (управляет вызовом операций по работе с объектами: редактор (поле TEditor), процессор (поле TProc), память (поле TMemory), буфер обмена (глобальный объект ClipBoard)), операция получает целое число (номер команды пользователя), строку для буфера обмена, строку со значением состояния памяти и возвращает строку для буфера обмена, строку состояния памяти и строку результата;
* “выполнитьКомандуРедактора” (управляет вызовом методов объекта редактор (тип TEditor)), операция получает целое число (номер команды пользователя и возвращает строку результата;
* “выполнитьОперацию” (управляет вызовом методов объекта процессор (поле TProc)), операция получает целое число (номер команды пользователя и возвращает строку результата;
* “выполнитьФункцию” (управляет вызовом методов объекта процессор (поле TProc)), операция получает целое число (номер команды пользователя и возвращает строку результата;
* “вычислитьОперацию” (управляет вызовом методов объекта процессор (поле TProc)), операция получает целое число (номер команды пользователя и возвращает строку результата;
* “вычислитьВыражение” (управляет вызовом методов объекта процессор (поле TProc)), операция получает целое число (номер команды пользователя и возвращает строку результата;
* “установитьНачальноеСостояниеКалькулятора” (управляет вызовом методов для перевода объекта типа TCalc в состояние **Start**(см. ниже), операция получает целое число (номер команды пользователя и возвращает строку результата;
* “выполнитьКомандуПамяти” (управляет вызовом методов объекта типа **TCtrl**, обеспечивающих выполнение команд памяти), операция получает целое число (номер команды пользователя), строку со значением состояния памяти и возвращает строку состояния памяти и строку результата;
  + “очистить память” (обеспечивает выполнение команды MC памяти), операция получает строку со значением состояния памяти и возвращает строку состояния памяти и строку результата;
  + “записать в память” (обеспечивает выполнение команды MS памяти), операция получает строку со значением состояния памяти и возвращает строку состояния памяти и строку результата;
  + “считать из памяти” (обеспечивает выполнение команды MR памяти), операция получает строку со значением состояния памяти и возвращает строку состояния памяти и строку результата;
  + “добавить к числу в памяти” (обеспечивает выполнение команды M+ памяти), операция получает строку со значением состояния памяти и возвращает строку состояния памяти и строку результата;
* “выполнитьКомандуБуфераОбмена” (управляет вызовом методов объекта типа **TClipBoard**, обеспечивающих выполнение команд буфера обмена), операция получает целое число (номер команды пользователя), строку со значением буфера обмена и возвращает строку со значением буфера обмена и строку результата;
  + “записатьВБуферОбмена” (обеспечивает выполнение команды Вставить), операция возвращает строку со значением числа и строку результата;
  + “читатьИзБуферОбмена” (обеспечивает выполнение команды Копировать), операция получает строку со значением числа и строку результата;
* “сменить основание системы счисления”, операция получает новое значение основания системы счисления и возвращает строку, содержащую отображаемое число в новой системе счисления;
* “читать | писать состояние калькулятора”, возвращает значение типа TCtrlState (свойство, опирающееся на поле);
* “конструктор”, осуществляет создание объектов и инициализацию полей класса;
* “деструктор”, осуществляет освобождение памяти, занимаемой объектом класса и объектами, указатели на которые хранятся в полях объекта: “Редактор”, “Процессор”, “Память”, “Число”.

Рекомендации к выполнению

Логика работы объекта “управление калькулятором” класс TCtrl может быть описана с помощью таблицы переходов, которая отражает изменение состояния калькулятора и результат работы объекта под действием команд пользователя. Таблица переходов строится на основе анализа прецедентов (вариантов использования), приведённых в спецификации.

Для построения таблицы переходов необходимо:

1. проанализировать спецификацию, приведённую в задании для калькулятора р-ичных чисел и выделить состояния калькулятора, например:**Start (Начальное), Editing (Ввод и редактирование), ExpDone (Выражение вычислено), FunDone (Функция выполнена), ValDone (Значение введено)**,**OpChange (смена операрации), Error (ошибка);**
2. построить таблицу состояний следующего вида:

|  |  |  |  |
| --- | --- | --- | --- |
| **Состояние калькулятора: Start (Начальное – после запуска приложения или выполнения команды С)** | | | |
| Вход (команда пользователя) | Результирующее состояние | Действия | Примечание |
| Команды ввода и редактирования | Editing | Изменение состояния редактируемой строки. (Вызов редактора). |  |
| Операция (Op) | Start | Введённая операция становится текущей. |  |
| Функция (Fun) | Start | Действия отсутствуют. |  |
| Error |
| Команды памяти | Start | Действия отсутствуют. |  |
| Команды буфера обмена | Start | Действия отсутствуют. |  |
| Вычислить (=) | Start | Действия отсутствуют. |  |
| Новое выражение (С) | Start | Действия отсутствуют. |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Состояние калькулятора: Editing (Редактирование)** | | | |
| Вход (команда пользователя) | Результирующее состояние | Действия | Примечание |
| Команды ввода и редактирования |  |  |  |
| Операция (Op) |  |  |  |
|  |
| Функция (Fun) |  |  |  |
|  |
| Взять из памяти (MR), (MS),(M+) |  |  |  |
| Вставить из буфера обмена, Копировать в буфер обмена |  |  |  |
| Вычислить (=) |  |  |  |
|  |
| Новое выражение (С) |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Состояние калькулятора: OpChange (смена операрации)** | | | |
| Вход (команда пользователя) | Результирующее состояние | Действия | Примечание |
| Команды ввода и редактирования |  |  |  |
| Операция (Op) |  |  |  |
| Функция (Fun) |  |  |  |
|  |
| Взять из памяти (MR) |  |  |  |
| Вставить из буфера обмена |  |  |  |
| Вычислить (=) |  |  |  |
|  |
| Новое выражение (С) |  |  |  |

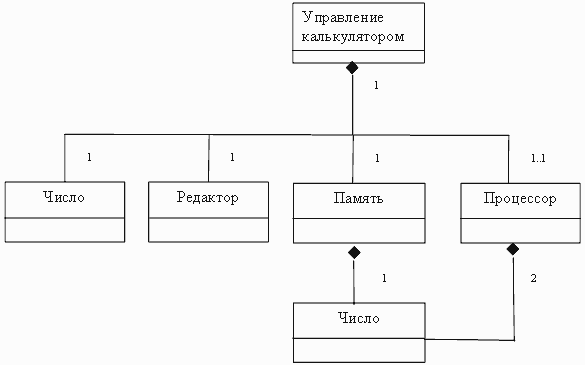
|  |  |  |  |
| --- | --- | --- | --- |
| **Состояние калькулятора: ExpDone (Выражение вычислено)** | | | |
| Вход (команда пользователя) | Результирующее состояние | Действия | Примечание |
| Команды ввода и редактирования |  |  |  |
| Операция (Op) |  |  |  |
| Функция (Fun) |  |  |  |
|  |
| Взять из памяти (MR) |  |  |  |
| Копировать из буфера обмена (MR) |  |  |  |
| (MS),(MC),(M+), Вставить в буфер обмена |  |  |  |
| Вычислить (=) |  |  |  |
|  |
| Новое выражение (С) |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Состояние калькулятора: FunDone (Функция вычислена),)** | | | |
| Вход (команда пользователя) | Результирующее состояние | Действия | Примечание |
| Команды ввода и редактирования |  |  |  |
| Операция (Op) |  |  |  |
|  |
| Функция (Fun) |  |  |  |
|  |
| Взять из памяти (MR) |  |  |  |
| Копировать из буфера обмена (MR) |  |  |  |
| (MS),(MC),(M+), Вставить в буфер обмена |  |  |  |
| Вычислить (=) |  |  |  |
|  |
| Новое выражение (С) |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Состояние калькулятора: ValDone (Значение введено)** | | | |
| Вход (команда пользователя) | Результирующее состояние | Действия | Примечание |
| Команды ввода и редактирования |  |  |  |
| Операция (Op) |  |  |  |
|  |
| Функция (Fun) |  |  |  |
|  |
| Взять из памяти (MR) |  |  |  |
| Копировать из буфера обмена (MR) |  |  |  |
| (MS),(MC),(M+), Вставить в буфер обмена |  |  |  |
| Вычислить (=) |  |  |  |
|  |
| Новое выражение (С) |  |  |  |

1. На основе таблицы состояний постройте таблицу переходов для калькулятора.

Диаграмма классов для Управления калькулятором р-ичных чисел представлено на рисунке ниже.



Содержание отчета

1. Задание.
2. Текст программы.
3. Тестовые наборы данных для тестирования класса.

Контрольные вопросы

1. Что такое инкапсуляция?
2. Как синтаксически представлено поле в описании класса?
3. Как синтаксически представлен метод в описании класса?
4. Как синтаксически представлено простое свойство в описании класса?
5. Особенности описания методов класса?
6. Особенности описания и назначение конструктора класса?
7. Видимость идентификаторов в описании класса?
8. Особенности вызова методов применительно к объектам класса?