**Лекция 4.**

**Классы общего назначения**

Библиотека MFC (Microsoft Foundation Classes) включает большое количество классов, многие из которых связаны друг с другом механизмом наследования.

Классы MFC можно разделить на две группы: общего назначения и предназначенные для создания интерфейса пользователя .

Классы общего назначения:

CString – последовательность символов, для которой определены некоторые методы и операции;

CFile – средство для работы с файлами на диске.

Классы для создания пользовательского интерфейса:

CDialog – диалоговые панели,

CButton – кнопки,

CStatic – статические строки,

CEdit – окна для редактирования текста.

**Класс CString**

Класс CString – удобное средство для работы со строками. В отличие от языка C, где работа со строковыми данными сводится к использованию массивов символов, ограниченных конечным нулем, а действия над строками осуществляются через указатели на строки, класс CString позволяет создать строковую переменную. CString не имеет базового класса.

Класс CString содержит последовательность символов переменной длины и набор функций и операций над ней.

В класс CString включено несколько конструкторов, в том числе конструктор по умолчанию, конструктор с параметрами, копирующий конструктор, при создании строк нет необходимости заботиться о выделении достаточного объема памяти для них, выделение памяти производится автоматически.

Примеры конструирования строковых объектов:

*CString str;* //Конструктор по умолчанию создает пустую строку

*CString str1,str2;//*Конструктор по умолчанию создает две пустые строки

*CString str3("character string ");* //Конструктор с параметрами

//помещает в строку str3 значение своего аргумента

*CString str4=str3;* //Создание строки и ее инициализация заданным значением

*CString str5 = "123456";* // Создание строки и ее инициализация заданным значением

*CString str6(str5);* //Копирующий конструктор создает копию переменной

//типа CString

В класс CString включены операции над строками.

Присваивание:

*str=" string ";*

*str1=str;*

Конкатенация строк (сложение):

*str1=" library ";*

*str2="MFC";*

*str=str1 + " " + str2;*

Доступ к отдельному символу строки через его порядковый номер:

Следует заметить, что для выполнения действий над отдельными символами лучше применять специальные функции класса GetAt() и SetAt():

*str.GetAt(1);* //Чтение символа с индексом 1,

*str.SetAt(2,'7');* //Запись символа '7' вместо символа с индексом 2.

Сравнение строк (используются перегруженные операторы отношения <, >, <=, >=, ==, !=):

*if (str1>str2)*

*str=str1;*

*else*

*str=str2;*

Вставка объекта в поток (оператор <<):

*cout << str1;*

Извлечение объекта из потока (оператор >>):

*cin >> str1;*

В класс CString включены разнообразные функции для обработки строковых данных.

|  |  |
| --- | --- |
| **Методы класса CString** | **Выполняемые действия** |
| GetLength() | Возвращает число символов, хранимых в объекте |
| GetBuffer() | Возвращает указатель на буфер объекта, что позволяет использовать его в обычных функциях языка C, имеющих в качестве параметра указатель на строку |
| ReleaseBuffer() | Освобождает буфер, распределенный Выполняемые действияметодом GetBuffer() |
| Compare() | Сравнивает две строки с учетом регистра букв |
| CompareNoCase() | Сравнивает две строки без учета регистра букв |
| GetAt() | Возвращает символ в заданной позиции |
| SetAt() | Устанавливает символ в заданную позицию |
| Find() | Находит подстроку и возвращает индекс ее первого символа |
| Format() | Преобразует данные других типов в текст, подобно функции sprintf() |
| MakeLower() | Преобразует все символы в строчные |
| MakeUpper() | Преобразует все символы в прописные |
| MakeReverse() | Обращает порядок символов в строке |
| Left() | Выделяет заданное количество самых левых символов в строке |
| Right() | Выделяет заданное количество самых правых символов в строке |
| Delete() | Удаляет указанные символы из строки |

Преобразование данных в переменную типа CString:

*CString str=" number";*

*str.Format("%s %d",str,123);*

В переменную str записывается слово. Функция Format имеет три параметра: первый представляет собой строку формата, остальные – список помещаемых в строку значений. Спецификаторы преобразования строки формата должны соответствовать типу соответствующих им аргументов. Так, первый спецификатор %s соответствует строковой, второй %d – целочисленной переменным. В результате преобразования в переменной str будет находиться строка: " number 123". При применении других методов класса CString следует обратиться к контекстной подсказке VC++ или к справочной системе MSDN.