# **ПКШ2021\_ЛР2. Разработка класса *DBDate* для представления даты в макете СУБД**

# **Цель работы**

Целью работы является приобретение студентами навыков создания и использования в программах на C++ собственных типов данных.

Задачи, решаемые при выполнении лабораторной работы:

* Проектирование класса (разработка UML-диаграммы класса)
* Разработка тестирующей программы testDBDate
* Перегрузка конструкторов
* Разработка служебных ( закрытых) функций
* Перегрузка операторов в виде членов класса
* Перегрузка операторов в виде дружественных функций

Планируемое время выполнения работы- 6 часов занятий в компьютерном зале + 3 часа самостоятельной работы студента (СРС).

# **2. Порядок выполнения работы**

*2.1. Проектирование* *интерфейса класса*, которое включает определение общедоступных функций-членов класса, которые должны выполняться объектами класса в приложениях, для которых разрабатывается класс.  
2.2. Проектирование состава параметров.

Параметры должны обеспечивать реализацию интерфейса класса.   
Совокупность значений параметров определяет состояние объекта класса.  
Функции-члены класса могут изменять состояние объекта или возвращать информацию о его состоянии.

2.3. Определение (программирование) методов класса.

Перегрузка операторов - это всего лишь другой способ вызова функций. Следовательно, перегружать операторы стоит только тогда, когда перегрузка заметно упрощает написание или, что еще важнее, чтение программ с вашим.   
В случае с классом DBDate это условие будет выполняться.

Одновременно с программированием методов класса должно выполняться и их тестирование. Для этого в проекте ЛР2 одновременно должны разрабатываться два файла:  
- файл testDBDateN.cpp с тестирующей программой, которая включает функцию main(),   
- файл DBDateN.cpp, в котором определяются методы класса.  
 Здесь N – индекс бригады разработчиков. Например, для бригады №1 везде должно использоваться имя класса DBDate1.  
В дальнейшем в описании ЛР для ссылки на разрабатываемый класс будет использоваться имя DBDate1. (В подключаемой к проекту библиотеке учебной СУБД имя класса DBDate.)

2.3.1. Разработка «скелета» тестирующей программы testDBDate1

- Создайте проект testDBDate1, аналогичный проекту testDB из ЛР1, заменив в основной функции testDBDate1.cpp меню на приведённое ниже меню «Разработка и тестирование класса DBDate1».

- Добавьте в проект файлы и dbmsLib1.h.   
 Файл DBDate1.cpp – пустой. В нём вы будете определять методы разрабатываемого класса DBDate1.  
 Файл dbmsLib1.h – это заголовочный файл для разрабатываемого класса DBDate1.  
Скопируйте в него из файла dbmsLib\_v1\_1.h всё, что относится к библиотечному классу DBDate, заменяя DBDate на DBDate1.

Не забудьте проверить правильность создания проекта, выполнив из п.8 меню функцию

void testing(){

string str2(" 29.2.2021 ");

dbmsLib1::DBDate1 datStr2;

cout<<"str2="<<str2<<" dat2="<<datStr2<<endl;

cout<<endl;

dbmsLib::DBDate datStr;

cout<<"str2="<<str2<<" dat2="<<datStr<<endl;

cout<<endl;

}

2.3.2. Дальнейшую разработку методов класса и тестирующей программы выполняйте параллельно в порядке, соответствующем пунктам меню 1, 2, 3, 4 и 5.

Правильность разработки методов класса DBDate1 проверяйте в тестирующей программе путём сравнения результатов выполнения аналогичных операций с объектами класса DBDate из библиотеки учебной СУБД dbmsLib\_v1\_1.

!!! При разработке программы используйте приведённые в п.3 Методических указаний источники информации.

int menu(){  
 cout<<"============Разработка и тестирование класса DBDate1 ===================\n";  
cout<<"\t1 - Создание объектов DBDate1 и проверка конструкторов\n";  
cout<<"\t2 - Перегрузка и тестирование логических операторов сравнения\n";  
cout<<"\t3 - Перегрузка и тестирование оператора DBDate1& operator+=(int days)\n"; cout<<"\t4 - Перегрузка и тестирование оператора DBDate1& operator-=(int days)\n"; cout<<"\t5 - Перегрузка и тестирование оператора int operator-(DBD1ate& date)\n"; cout<<"\t8 - Тестирование\n";  
cout<<"\t10 - Выход\n";  
int choice;  
cout<<"Выберите действие\n";  
cin>>choice;  
while(cin.fail()){  
 cout<<"Ошибка ввода. Повторите ввод\n";  
 cin.clear();  
 cin.ignore(10,'\n');  
 cin>>choice;  
 }  
return choice;  
}

//-----------------класс DBDate1----------------------------

class DBDate1

{

static const int arrDaysInMonths[13];

int GetDaysInMonth(int month,int year);//Количество дней в месяце

friend string DateToStr1(DBDate1& date);

friend ostream& operator<<(ostream& out,DBDate1& date);

int day, month, year;

bool IsCorrectDate(int d, int m, int y);

public:

DBDate1(string date);//формат строки: dd.mm.yyyy

DBDate1(int d,int m,int y);

DBDate1():day(1),month(1),year(1){};//конструктор по умолчанию

DBDate1(DBDate1& dat):day(dat.day),month(dat.month),year(dat.year){}

int GetDay();

int GetMonth();

int GetYear();

bool IsLeapYear (int year); // год високосный?

int DaysInCurYear();//Количество дней от начала года до текущей даты

bool operator==(DBDate1& date);

bool operator<(DBDate1& date);

bool operator>(DBDate1& date);

bool operator<= (DBDate1& date);

bool operator>= (DBDate1& date);

bool operator!= (DBDate1& date);

DBDate1& operator+=(int days);//Прибавляет к текущей дате days дней

DBDate1& operator-=(int days);//Вычитает из текущей даты days дней

int operator-(DBDate1& date);//Количество дней между текущей датой и date

//Если текущая дата > date, результат < 0.

};

Рекомендуемые источники информации

1. Лафоре Р. Объектно-ориентированное программирование в С++. 4-е изд.- СПб.: Питер, 2004.- 922с.: Глава 6. Объекты и классы, Глава 8 «Перегрузка операций»

1. Эккель Б. Философия С++. Введение в стандартный С++. 2-е изд.- СПб.: Питер, 2004.- 572с.: ил. Глава 12 «Перегрузка операторов»

3. Эккель Б. Ч. Эллисон Философия С++. Практическое программирование. 2-е изд.- СПб.: Питер, 2004.- 608с.: ил. Глава 1. Обработка исключений, Глава 3. Строки.