

КЕМЕРОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНСТИТУТ ФУНДАМЕНТАЛЬНЫХ НАУК
КАФЕДРА ЮНЕСКО ПО ИНФОРМАЦИОННЫМ ВЫЧИСЛИТЕЛЬНЫМ
ТЕХНОЛОГИЯМ

ЛАБОРАТОРНАЯ РАБОТА № 1

по дисциплине «*Языки программирования*»
Семестр 3

Тема: *Структуры*

Выполнил:

студент группы _____

(Фамилия И.О.)

Проверил:

_____кафедры ЮНЕСКО по ИВТ

(должность)

(степень, звание)

(Фамилия И.О.)

Кемерово, 2020

Регламент сдачи лабораторной работы:

Файлы исходного кода должны выкладываться в открытом репозитории на *GitHub.com*. При отсутствии выложенных файлов лабораторная работа не принимается.

Отчет о выполнении лабораторной работы сдается через систему ИнфОУПро. Отчет должен включать титульный лист, формулировку задания, примеры тестирования программы со скриншотами запуска и URL-ссылку на репозиторий на *GitHub.com*.

Защита лабораторной работы выполняется устно. Для этого необходимо продемонстрировать преподавателю лабораторных занятий работу программы на тестовых наборах данных. Нужно уметь объяснять принцип работы программы.

Назначаются контрольные даты. Лабораторная работа должна быть сдана до назначенной контрольной даты. Если лабораторная работа сдается с недельным опозданием, то от полученных баллов отнимается 25%. Сданные позднее оговоренного срока лабораторные работы принимаются со штрафом в 50%.

Реализуйте следующие программы на языке C++.

Задание:

1. Необходимо описать объект «Дата» с помощью структуры с именем `datetime` и следующими полями: `second`, `minute`, `hour`, `day`, `month`, `year`.
2. Для структуры предусмотреть методы: вывод информации, получение следующей даты, получение предыдущей даты.
3. В главной программе необходимо создать массив из n объектов. Инициализировать массив данными, считанными из файла.

Вывести информацию о всех объектах на экран в формате

`<номер> <дата> <следующая дата> <предыдущая дата>`

Количество записей в файле неизвестно.

Задания по вариантам:

Вариант 1:

Выбрать из массива исходных дат те, для которых следующие попадут на другой месяц. Вывести информацию о таких датах на экран в формате:

`<номер> <дата> <следующая дата>`

Вариант 2:

Выбрать из массива исходных дат те, для которых следующие попадут на другой год. Вывести информацию о таких датах на экран в формате:

`<номер> <дата> <следующая дата>`

Вариант 3:

Выбрать из массива исходных дат те, для которых следующие не попадут на другой месяц. Вывести информацию о таких датах на экран в формате:

`<номер> <дата> <следующая дата>`

Вариант 4:

Выбрать из массива исходных дат те, для которых следующие не попадут на другой год. Вывести информацию о таких датах на экран в формате:

<номер> <дата> <следующая дата>

Вариант 5:

Выбрать из массива исходных дат те, для которых предыдущие попадают на другой месяц. Вывести информацию о таких датах на экран в формате:

<номер> <дата> <предыдущая дата>

Вариант 6:

Выбрать из массива исходных дат те, для которых предыдущие попадают на другой год. Вывести информацию о таких датах на экран в формате:

<номер> <дата> <предыдущая дата>

Вариант 7:

Выбрать из массива исходных дат те, для которых предыдущие не попадают на другой месяц. Вывести информацию о таких датах на экран в формате:

<номер> <дата> <предыдущая дата>

Вариант 8:

Выбрать из массива исходных дат те, для которых предыдущие не попадают на другой год. Вывести информацию о таких датах на экран в формате:

<номер> <дата> <предыдущая дата>

Вариант 9:

Выбрать из массива исходных дат те, для которых следующие и предыдущие даты приходятся на один и тот же месяц. Вывести информацию о таких датах на экран в формате:

<номер> <дата> <следующая дата> <предыдущая дата>

Вариант 10:

Выбрать из массива исходных дат те, у которых номер дня нечетный, а номер дня следующей даты четный. Вывести информацию о таких датах на экран в формате:

<номер> <дата> <следующая дата> <предыдущая дата>

Вариант 11:

Выбрать из массива исходных дат те, у которых номер дня четный, и номер дня следующей даты четный. Вывести информацию о таких датах на экран в формате:

<номер> <дата> <следующая дата> <предыдущая дата>

Вариант 12:

Выбрать из массива исходных дат те, у которых номер дня нечетный, и номер дня следующей даты нечетный. Вывести информацию о таких датах на экран в формате:

<номер> <дата> <следующая дата> <предыдущая дата>

Вариант 13:

Выбрать из массива исходных дат те, у которых номер дня четный, а номер дня следующей даты нечетный. Вывести информацию о таких датах на экран в формате:

<номер> <дата> <следующая дата> <предыдущая дата>

Вариант 14:

Выбрать из массива исходных дат те, для которых следующие и предыдущие даты приходятся на один и тот же год. Вывести информацию о таких датах на экран в формате:

<номер> <дата> <следующая дата> <предыдущая дата>

Задание повышенной сложности

1. Для структуры `datetime` реализовать метод, который возвращает текущую системную дату и время в виде структуры `datetime`.
2. Описать объект «Промежуток времени» с помощью структуры с именем `timedelta` и следующими полями: `second`, `minute`, `hour`, `day`, `month`, `year`. (Промежуток времени может иметь отрицательные значения полей)
3. Для данных структур реализовать функции, выполняющие следующие действия:

`datetime1 - datetime2 = timedelta`

`datetime1 +- timedelta = datetime2`

`timedelta1 +- timedelta2 = timedelta3`

Требования к оформлению программ:

1. **Функциональная декомпозиция кода.** Программа должна быть оформлена с помощью функций. В главной функции реализовывается только управляющий алгоритм, вспомогательные алгоритмы должны быть реализованы в дополнительных функциях.
2. **Тестирование.** Программа должна быть протестирована на разнообразных входных данных, в том числе и тех, которые могут привести к ошибочному поведению. Код программы должен содержать достаточное количество проверок на корректность входных данных.

Критерии оценки (максимально 20 баллов):

В случае, если в задаче получен верный результат, то за решение начисляются баллы.

Базовое задание:

Правильно используются структуры для описания пользовательского типа данных в соответствии с заданием.	2
Правильно реализована работа с внешними файлами данных	1
Правильно реализована работа с динамическими массивами структур. Памяти выделяется необходимое количество. Нет утечки памяти.	2
Правильно находятся следующие даты.	1
Правильно находятся предыдущие даты.	1
Правильно реализована выборка.	3

Задание повышенной сложности:

Правильно реализован метод, который возвращает текущую системную дату и время в виде структуры datetime	2
Правильно реализованы функции для работы со структурами datetime и timedelta	8

Срок сдачи: две недели.