

Обектно-ориентирано програмиране

Самостоятелна работа, 30.03. – 03.04.2020 г.

Задача 1.

Да се дефинира клас Служител, който описва служител на дадена фирма със следните характеристики:

- име – символен низ с произволна дължина;
- отработени часове – положително реално число;
- заплащане на час – реално число, което е по-голямо от 3.50 лв.

Да се дефинират подходящи конструктори, деструктор, подходящи селектори и мутатори.

Даден е двоичен файл с информация за отработените от служителите на фирмата часове за текущата седмица. Този двоичен файл да се трансформира до *текстов* файл със следния формат на данните:

Иван Иванов	20.5	5.40
Детелина Горанова	40.50	8.60
Георги Огнянов	36	20.50

Да се генерира *текстов* файл, в който са изчислени заплатите на служителите за текущата седмица, на базата на файла с отработените часове. След обработването на файла, на екрана да се изведе информация за общата сума, изплатена за заплати.

Да се изведат в текстов файл всички служители, които са работили по-малко от 20 часа седмично.

Задача 2.

A. Да се дефинира клас Ticket, описващ самолетен билет с характеристики:

- име на пътник – символен низ с произволна дължина, записан в динамичната памет;
- номер на полет – точно 6 символа, първите два от които са главни латински букви (от A до Z), последвани от 4 цифри;
- място – точно 3 символа, първите два от които съответстват на номера на реда (от 1 до 25), а последният съответства на мястото, отбелязано с главна латинска буква от A до F.

Да се дефинират подходящи конструктори, селектори и мутатори.

Да се направи проверка за валидност на данните.

B. Да се дефинира клас Flight, описващ самолетен полет с характеристики:

- номер на полет – точно 6 символа, първите два от които са главни латински букви (от A до Z), последвани от 4 цифри;
- дата и час на полет – символен низ с дължина точно 14, записан във формат “ууууmmdd hh:mm”. Датата и часът на полета са разделени с интервал;
- масив от билети, продадени за полета. Броят на билетите, продадени за полета не може да надхвърля 150.

Да се дефинират следните член-функции:

- bool boarding (constTicket&),
която проверява дали указаният като аргумент билет е за текущия полет. За сравнение да се използва номерът на полета. Ако условието е изпълнено, билетът се записва в масива от билети на полета. Позицията, на която трябва да бъде записан билетът се определя от полето място на билета. Например билет с номер 1A се записва на позиция 0, билет с номер 2A се записва на позиция 6, ..., билет с номер 25A се записва на

позиция 144. Ако билетът не е за текущия полет, функцията връща лъжа, без да променя информацията за текущия полет.

- `size_t passengersOnBoard() const`,
която връща като резултат броя на пътниците, които са на борда на самолета.

Да се дефинира функция `main()`, в която:

- да се създаде полет с номер AF3295, който излита на 15.04.2018 в 06:20;
- да се качат на самолета пътниците с имена Иван Иванов и Мария Иванова, съответно на места 23A и 23B;
- да се изведе информация за броя на пътниците, които са на борда.