

## Первый раздел курсовой работы по дисциплине ООП\_2018

### Вариант Е-1

Создать иерархию классов. Базовый класс – очередь с двумя концами. Элементы могут удаляться и добавляться с любой стороны. Создать производные классы – стек и очередь (обычная).

### Вариант Е-2

Есть три абстракции: линия (бесконечная), луч (бесконечен в одну сторону) и отрезок (ограничен). Разработать иерархию классов.

### Вариант Е-3

Реализовать алгоритм сортировки массива по возрастанию. Метод сортировки – пузырьковая сортировка. Пузырьковая сортировка основана на методе перестановок. Ее смысл заключается в постоянном сравнении смежных элементов и при необходимости их перестановки.

### Вариант Е-4

Создать иерархию классов. Создать класс студент, определить его имя, курс и идентификационный номер. Создать public-производный класс – студент-дипломник, имеющий тему диплома. Определить конструкторы по умолчанию и конструкторы с разным числом параметров. Определить функции переназначения названия диплома и идентификационного номера.

### **Вариант Е-5**

Используя иерархию и композицию классов, создать бинарное дерево. У бинарного дерева есть корневой узел. Мы можем вставлять узел. Мы можем обходить вначале левое поддерево, потом правое (последовательный обход) и обратный обход. Узел может быть помещен в дерево двоичного поиска только в качестве конечного узла. Если дерево является пустым, то создается новый экземпляр класса узел дерева и узел помещается в дерево. Если дерево не является пустым, то программа сравнивает вставляемое в дерево значение со значением в корневом узле и если меньше, то помещает в левые поддеревья, а если больше, то в правые. Если значения равны, то выводится сообщение, что повтор и не вставляется.

### **Вариант Е-6**

Создать иерархию классов. Создать класс строка и производный от него – класс строк из цифр. Определить конструкторы. Обеспечить запись строки в текстовый файл. Перегрузить оператор присваивания и конструктор копирования в базовом и производном классе.

### **Вариант Е-7**

Создать классы геометрических фигур: четырехугольников, квадратов и прямоугольников. Создать из них иерархию. Определить функции ввода-вывода, вычисление площади геометрических фигур и периметра.

### **Вариант Е-8**

Написать параметризованную функцию сортировки *методом отбора*. Сортировка методом отбора заключается в следующем: алгоритм находит элемент с наименьшим значением и выполняет перестановку,

меняя его местами с первым элементом массива. После этого из оставшихся  $n-1$  элементов ищется наименьший, после чего осуществляется его перестановка со вторым элементом и так далее. Перестановка будет осуществляться до тех пор, пока не поменяются местами последние два элемента. Например, если бы строка "dcab" сортировалась методом отбора, то каждый из проходов давал бы результат:

Исходный массив	Dcab
Проход 1	Acdb
Проход 2	Abdc
Проход 3	Abcd

### Вариант Е-9

Написать параметризованную функцию сортировки методом быстрой сортировки.

Алгоритм быстрой сортировки построен на основе идеи разбиения массива на разделы. Общая процедура заключается в выборе пограничного значения, называемого *компарандом*, которое разбивает сортируемый массив на две части. Все элементы, значения которых больше пограничного значения, переносятся в один раздел, а все элементы с меньшими значениями — в другой. Затем этот процесс повторяется для каждой из частей и так до тех пор, пока массив не будет отсортирован. Например, допустим, что необходимо отсортировать следующий массив: "fedacb". Если в качестве компарадера использовать "d", то после первого прохода алгоритм быстрой сортировки упорядочит массив следующим образом

Исходный массив	Fedacb
Проход 1	Bcdef

Затем этот подход повторяется для каждого раздела, а именно "bca" И

"def".

Проход 2	Acb	Def
Проход 3	Abc	Def

Процесс рекурсивен по своей природе и наилучшие решения получаются при рекурсивном подходе. Пограничное значение можно выбирать двумя путями. Во-первых, его можно делать случайным образом, или путем осреднения небольшого набора значений, принадлежащих к разделу. Для того, чтобы сортировка была оптимальной, надо выбирать значение, расположенное точно в середине диапазона значений. Поскольку на практике это не всегда осуществимо, можно выбирать срединный элемент каждого из разделов.

#### **Вариант E-10**

Написать параметризованную функцию сортировки методом вставки. В методе вставки на первом шаге выполняется сортировка первых двух элементов массива. Далее алгоритм ставит третий элемент в порядковую позицию, соответствующую его положению относительно первых двух элементов. Затем в этот список вставляется четвертый элемент и т.д. Процесс продолжается до тех пор, пока все элементы не будут отсортированы. Например, если бы строка "dcab" сортировалась методом вставки, то каждый из проходов давал бы результат:

Исходный массив	Dcab
Проход 1	Cdab
Проход 2	Acdb
Проход 3	Abcd

#### **Вариант E-11**

Написать параметризованную функцию сортировка методом Шелла.

Метод построен на основе метода вставки с минимизацией промежуточных шагов. Рассмотрим таблицу

Проход1	F	D	A	C	B	E
Проход1	C	B	A	F	D	E
Проход1	A	B	C	F	D	E
Результат	A	B	C	D	E	F

Сначала выполняется сортировка элементов, отстоящих друг от друга на три позиции. После этого сортируются элементы, отстоящие друг от друга на две позиции. Наконец, выполняется сортировка смежных элементов. Точная последовательность изменения приращений может изменяться. Единственным требованием остается равенство последнего приращения 1. Например, хорошо себя зарекомендовала последовательность 9, 5, 3, 2, 1, которую и предлагается использовать в контрольном примере.

### **Вариант Е-12**

Написать функцию-шаблон бинарного поиска. Если данные, по которым требуется провести поиск, отсортированы, то можно использовать бинарный поиск. Поэтому, первоначально надо написать функцию сортировки. При использовании бинарного метода на первом шаге проверяется срединный элемент. Если он больше ключа поиска, то проверяется срединный элемент второй половины массива. Эта процедура повторяется до тех пор, пока не будет найдено совпадение, или до тех пор, пока больше не останется элементов, которые можно было бы проверять.

### **Вариант Е-13**

Создать параметризованный класс бинарного дерева с методами.

Добавить элемент в дерево, прохождение по дереву в нисходящем и в восходящем порядке. Осуществить поиск по дереву.

#### **Вариант Е-14**

Создать параметризованный тип данных – множество. Этот тип предназначен для хранения множества элементов и выполнения операций над ними. Реализовать классические операции над множествами: объединение, пересечение и симметричная разность. Для создания данного типа необходимо иметь простые представления о теории множеств.

#### **Вариант Е-15**

Создать класс матрица. Данный класс содержит размер строк и столбцов матрицы. Определить конструктор без параметров, конструктор с одним параметром и конструктор с двумя параметрами, деструктор. Определить методы доступа: возвращать значение элемента  $(i, j)$  и адрес этого элемента. Определить функции сложения и вычитания (матрицы с матрицей), умножение матрицы на матрицу. Определить умножение матрицы на число. Проверить работу этого класса.

#### **Вариант Е-16**

Создать иерархию классов для описания автомобилей. Создать класс машина, в котором определить марку, число цилиндров, мощность. Создать public-производный класс – грузовик, имеющий грузоподъемность кузова. Определить конструкторы по умолчанию и конструкторы с разным числом параметров. Определить функции переназначения марки и грузоподъемности.

### **Вариант E-17**

Создать иерархию классов мебель. Определить производные классы: стол, стул, диван и т. д. Переопределить вывод в файл и ввод данных из файла, конструктор копирования, оператор присваивания через соответствующие функции базового класса.

### **Вариант E-18**

Создать класс колесо, имеющий радиус. Определить конструкторы и метод доступа. Создать класс машин, содержащий класс колесо. Дополнительно определить марку машины и цену. Определить конструкторы. Определить public- производный класс грузовик, имеющий дополнительно грузоподъемность. Определить конструкторы, деструкторы и вывод в файл.

### **Вариант E-19**

Создать класс квартира. Определить основные свойства квартиры: количество комнат, общая площадь, этаж и др. Определить конструктор и метод доступа. Создать класс однокомнатных квартир, содержащий комнату и кухню (и их площадь), этаж (комната содержится в классе однокомнатная квартира). Определить другие типы квартир. Определить конструкторы, методы доступа. Определить public-производный класс однокомнатных квартир разных районов (дополнительный параметр – название района).

### **Вариант E-20**

Цифровой счетчик – это переменная с ограниченным диапазоном, которая сбрасывается, когда ее целочисленное значение достигает

определенного максимума. Примеры: цифровые часы, счетчик километража. Опишите класс такого счетчика. Обеспечьте возможность установления максимального и минимального значений, увеличения значений счетчика на 1, возвращения текущего значения. Опишите производные классы цифровые часы и счетчик километража.

#### **Вариант Е-21**

Создать класс колоду карт. Создать конструкторы колоды должны инициализировать колоду как случайным образом, так и в определенном порядке. Создать производный класс от колоды - пасьянс, в котором выбираются по три карты и, если две крайние одного цвета, то их выбрасывают. Всю колоду проходят три раза.

#### **Вариант Е-22**

Создать класс животное. Обеспечить классификацию животных: дикие, домашние. Создать public-производный класс - домашнее животное, имеющий кличку. Определить конструкторы по умолчанию и с разным числом параметров. Определить функции переназначения названия клички и числа потомков.

#### **Вариант Е-23**

Разработать программу, решающую задачу о 8 ферзях. На шахматной доске расставить 8 ферзей таким образом, чтобы ни один не бил другого. Изменить программу так, чтобы она выдавала все возможные решения. Сколько существует различных решений. Сколько из них являются поворотами других? Как избежать поворотов?

#### **Вариант Е-24**

Создать класс карта, имеющая ранг и масть. Карту можно перевернуть и



открыть. Создать класс - колода карт, содержащий карты. Создать два производных класса от колоды карт, в одном карты могут доставаться только по порядку, в другом - вытаскиваться произвольно.

#### **Вариант E-25**

Создать класс транспортные средства. Определить классы автомобиль, грузовик, пароход и самолет. Создать из них иерархию, используя механизмы наследования.

#### **Вариант E-26**

Создать класс окно, используя композицию и иерархию, определить окно с заголовком и кнопкой. Класс заголовков и кнопка тоже должны быть определены. Навесить на кнопку функцию обратного вызова - (метод класса окна). Эта функция вызывается, когда нажимается любая клавиша.

#### **Вариант E-27**

Создать систему классов, описывающих домашних животных. Создать из них иерархию, используя механизмы наследования.

#### **Вариант E-28**

Создать класс человек, имеющий имя (указатель на строку), возраст, вес. Определить конструкторы, деструктор и запись в файл. Создать public-производный класс - совершеннолетний, имеющий номер паспорта. Определить конструкторы по умолчанию и конструкторы с разным числом параметров, деструкторы, функцию вывода данных в файл. Определить функции переназначения возраста и номера паспорта.

### **Вариант E-29**

Разработать систему классов для реализации различных типов графов и операций над ними.

### **Вариант E-30**

Создать систему классов для обеспечения работы с абонентами телефонной компании.

### **Вариант E-31**

Разработать систему классов для обеспечения работы деканата.

### **Вариант E-32**

Разработать систему классов для обеспечения работы приемной комиссии университета.

### **Вариант E-33**

Построить систему классов для обеспечения работы библиотеки университета.

### **Вариант E-34**

Создать систему классов для описания плоских геометрических фигур: круга, квадрата, прямоугольника. Предусмотреть методы для создания объектов, перемещения на плоскости, изменения размеров и поворота на заданный угол.

### **Вариант E-35**

Разработать систему классов «университет». Представить организационную структуру университета. Применить механизмы наследования.

#### **Вариант Е-36**

Разработать систему классов, описывающих сотрудников предприятия/организации с их функциями (сотрудник, менеджер, инженер и т.д.).

#### **Вариант Е-37**

Разработать систему классов, описывающих городские транспортные средства: автобус, трамвай, троллейбус и т.д.

#### **Вариант Е-38**

Разработать систему классов, описывающих работу поликлиники.

#### **Вариант Е-39**

Разработать систему классов для представления легковых автомобилей с закрытыми кузовами (седан, лимузин, хетчбэк, универсал и т.д.).

#### **Вариант Е-40**

Разработать систему классов для представления спортивных и гоночных автомобилей.

#### **Вариант Е-41**

Разработать систему классов для представления железнодорожного транспорта.

#### **Вариант Е-42**

Разработать систему классов для представления работы аптеки.

#### **Вариант Е-43**

Разработать систему классов для представления работы продуктового магазина.

#### **Вариант Е-44**

Разработать систему классов для представления работы магазина спортивного оборудования.