

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Волхонов Михаил Станиславович
Должность: Ректор
Дата подписания: 23.12.2024 15:14:56
Уникальный программный ключ:
40a6db1879d6a9ee29ec8e0ffb2f95e4614a0998

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«КОСТРОМСКАЯ ГОСУДАРСТВЕННАЯ СЕЛЬСКОХОЗЯЙСТВЕННАЯ АКАДЕМИЯ»

Утверждаю:

И.о. декана электроэнергетического
факультета

Николай
Александрович
Климов

Подписано цифровой
подписью: Николай
Александрович Климов
Дата: 2024.09.11 16:07:39
+03'00'

/Климов Н.А./

11 сентября 2024 года

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
по дисциплине

ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

Специальность 09.02.07 Информационные системы и программирование

Квалификация выпускника программист

Форма обучения очная

Срок освоения ППССЗ 3 года 10 месяцев

На базе основного общего образования

Фонд оценочных средств предназначен для оценивания сформированности компетенций по дисциплине «Объектно-ориентированное программирование».

Разработчик: Алексей Сергеевич Яблоков Преподаватель кафедры информационных технологий в электроэнергетике А.С. Яблоков

Подписано цифровой подписью: Алексей Сергеевич Яблоков
Дата: 2024.09.05 14:45:32 +03'00'

Утвержден на заседании кафедры информационных технологий в электроэнергетике, протокол № 1 от 05.09.2024 года.

Заведующий кафедрой Н.А. Климов Николай Александрович Климов

Подписано цифровой подписью: Николай Александрович Климов
Дата: 2024.09.05 14:45:44 +03'00'

Согласовано: Алексей Сергеевич Яблоков председатель методической комиссии электроэнергетического факультета протокол № 7 от 10.09.2024 года.

А.С. Яблоков Алексей Сергеевич Яблоков

Подписано цифровой подписью: Алексей Сергеевич Яблоков
Дата: 2024.09.10 15:07:05 +03'00'

Результаты освоения дисциплины: «Объектно-ориентированное программирование»

ППССЗ (СПО) по специальности:

09.02.07 Информационные системы и программирование

Таблица 1

Коды компетенций по ФГОС	Компетенции	Результат освоения
Общие компетенции		
ОК 02.	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности	<p>Знать номенклатуру информационных источников, применяемых в профессиональной деятельности; приемы структурирования информации; формат оформления результатов поиска информации</p> <p>Уметь определять задачи для поиска информации; определять необходимые источники информации; планировать процесс поиска; структурировать получаемую информацию; выделять наиболее значимое в перечне информации; оценивать практическую значимость результатов поиска; оформлять результаты поиска</p> <p>Владеть навыками использования современных средств поиска, анализа и интерпретации информации; информационными технологиями для выполнения профессиональной деятельности</p>
Профессиональные компетенции		
ПК 1.2.	Разрабатывать программные модули в соответствии с техническим заданием	<p>Знать основные этапы разработки программного обеспечения; основные принципы технологии структурного и объектно-ориентированного программирования.</p> <p>Уметь создавать программу по разработанному алгоритму как отдельный модуль</p> <p>Владеть навыком разработки кода программного продукта на основе готовой спецификации на уровне модуля</p>
ПК 1.3.	Выполнять отладку программных модулей с использованием специализированных программных средств	<p>Знать основные принципы отладки и тестирования программных продуктов; инструментарий отладки программных продуктов.</p> <p>Уметь выполнять отладку и тестирование программы на уровне модуля. Применять инструментальные средства отладки программного обеспечения.</p> <p>Владеть навыком использования инструментальных средства на этапе отладки программного продукта</p>

ПК 1.6.	Разрабатывать модули программного обеспечения для мобильных платформ	<p>Знать основные этапы разработки программного обеспечения; основные принципы технологии структурного и объектно-ориентированного программирования.</p> <p>Уметь осуществлять разработку кода программного модуля на современных языках программирования.</p> <p>Владеть навыком разработки мобильных приложений</p>
---------	--	--

Требования к результатам освоения дисциплины:

иметь навык:

Н₁ – создания программных продуктов с использованием объектно-ориентированного подхода

уметь:

У₁ – создавать классы, объекты, выстраивать иерархию классов

знать:

З₁ – принципы объектно-ориентированного программирования

**Паспорт
фонда оценочных средств**

Таблица 2

№ п/п	Контролируемые дидактические единицы	Контролируемые компетенции (или их части)	Наименование оценочных средств		
			Тесты, кол-во заданий	Другие оценочные средства	
				вид	кол-во заданий
1	Раздел 1. Парадигма объектно-ориентированного программирования	ОК 02 ПК 1.2 ПК 1.3 ПК 1.6	37	Опрос	66
2	Раздел 2. Паттерны проектирование	ОК 02 ПК 1.2 ПК 1.3 ПК 1.6	-	Опрос	30
Всего:			37		96

**Методика проведения контроля по проверке базовых знаний по дисциплине
«Объектно-ориентированное программирование»**

Раздел 1. Парадигма объектно-ориентированного программирования

Контролируемые компетенции ОК 02; ПК 1.2; ПК 1.3; ПК 1.6

Вопросы для устного опроса

1. Что такое инкапсуляция?
2. Что такое класс в C++? Как выглядит описание класса?
3. Что такое поля и методы класса?
4. Какие существуют спецификаторы доступа к членам класса и какие функции они выполняют?
5. Возможна ли инициализация полей класса при их описании?
6. Что такое конструктор, какие функции он выполняет и как выглядит его описание?
7. Что такое деструктор, какие функции он выполняет и как выглядит его описание?
8. Что такое конструктор копирования, какие функции он выполняет и как выглядит его описание?
9. Как создать экземпляр класса?
10. С помощью каких операторов осуществляется доступ к элементам объекта?
11. Можно ли напрямую обратиться к закрытым полям объекта из внешнего кода программы?
12. Как объявить константный метод класса?
13. На что ссылается указатель this?
14. Что такое дружественные функции и для чего они используются?
15. Как объявить статические поля класса?
16. Как объявить статический метод класса и как его потом вызвать в коде программы?
17. Какие переменные хранятся в статической памяти?
18. Какие переменные хранятся в стеке?
19. В какой области памяти разработчик может выделять место по своей инициативе в ходе работы программы?
20. С помощью каких операторов и как осуществляется выделение и освобождение динамической памяти?
21. Как выглядит выделение и освобождение памяти для массивов?
22. Какие элементы класса можно использовать для автоматического выделения и освобождения динамической памяти при создании и удалении объекта?
23. Что такое полиморфизм?
24. Что такое перегрузка функций?
25. Чем должны отличаться перегруженные функции друг от друга?
26. Как объявляются аргументы по умолчанию в функции?
27. Где, по отношению к обычным аргументам функции, должны располагаться аргументы по умолчанию?
28. Как в общем виде выглядит перегрузка операторов?
29. Чем отличается перегрузка операторов префиксного и постфиксного инкремента?
30. Может ли оператор присваивания перегружаться с помощью дружественной функции класса?
31. Можно ли с помощью перегрузки изменить приоритет операторов?
32. Что такое композиция классов?
33. Что такое наследование?
34. Как выглядит описание производного класса?

35. Можно ли из производного класса получить доступ к полям базового класса, объявленного со спецификатором `protected`?
36. Наследуются ли конструктор и деструктор?
37. Что такое повышающее приведение типов?
38. На каком этапе жизни программы осуществляется сопоставление вызова функции с телом функции при позднем связывании?
39. Как сделать метод класса виртуальным?
40. Может ли деструктор быть виртуальным?
41. Может ли виртуальный метод объявляться с модификатором `static`?
42. Наследуется ли виртуальный метод?
43. Что такое абстрактный класс?
44. Можно ли создать объект абстрактного класса?
45. Как сделать виртуальный метод чисто виртуальным?
46. Должен ли переопределяться виртуальный метод внутри производного класса?
47. Что производный класс не наследует от базового класса?
48. В каком порядке вызываются конструкторы и деструкторы класса при создании и удалении объекта производного класса?
49. Предположим, что и в базовом, и в производном классе определен метод с одним и тем же именем, и производный класс вызывает этот метод. Какой метод будет вызван?
50. В каких случаях производный класс должен определять операцию присваивания?
51. Можно ли присвоить адрес объекта производного класса указателю на базовый класс? Можно ли присвоить адрес объекта базового класса указателю на производный класс?
52. Предположим, что определена функция, которая принимает в качестве аргумента ссылку на объект базового класса. Почему эта функция может также использовать в качестве аргумента объект производного класса?
53. Почему обычно лучше передавать объекты по ссылке, а не по значению?
54. Объясните разницу между виртуальными и не виртуальными базовыми классами
55. В чем состоит различие между `throw` и `return`?
56. Предположим, что имеется иерархия классов исключений, порожденная от базового класса исключений. В каком порядке следует расположить блоки `catch`?
57. Чем отличается операция `static_cast` от операции `dynamic_cast`
58. Назовите хотя бы два преимущества объектов `string` по сравнению со строками в стиле C с точки зрения простоты применения
59. Напишите функцию, которая принимает ссылку на объект `string` в качестве аргумента и затем преобразует объект `string` в прописные буквы
60. Приведите, как минимум, три примера, показывающих преимущества объекта `vector` по сравнению с обычным массивом.
61. Если указатель — это итератор, почему разработчики STL просто не используют его вместо итераторов?
62. Какую роль играет файл `iostream` во вводе-выводе C++?
63. В чем состоит различие между потоком стандартного вывода и стандартным потоком ошибок?
64. Почему `cout` может отображать различные типы C++ без необходимости явного указания инструкций для каждого типа?
65. Какое свойство определений методов вывода позволяет выполнять конкатенацию вывода?
66. Напишите программу, которая запрашивает целое число и затем отображает его в десятичной, восьмеричной и шестнадцатеричной формах.

Критерии оценки:

Оценка «отлично» выставляется обучающемуся, который прочно усвоил программный материал в полном объеме, исчерпывающе, грамотно и логически стройно его излагает, четко формулирует основные понятия, приводит соответствующие примеры, уверенно владеет материалом.

Оценка «хорошо» выставляется обучающемуся, который твердо усвоил программный материал, грамотно и по существу излагает его без существенных ошибок, правильно применяет теоретические положения при решении конкретных задач, с небольшими погрешностями приводит формулировки определений, по ходу изложения допускает небольшие пробелы, не искажающие содержания ответа.

Оценка «удовлетворительно» выставляется обучающемуся, который не совсем твердо владеет программным материалом, знает основные теоретические положения изучаемой темы, при ответах допускает малосущественные погрешности, искажения логической последовательности при изложении материала, неточную аргументацию теоретических положений, испытывает затруднения при ответе на дополнительные вопросы.

Оценка «неудовлетворительно» выставляется обучающемуся, имеющему серьезные пробелы в знании учебного материала, допускающему принципиальные ошибки при ответе на вопросы.

Тестовые задания

Выберите один правильный вариант ответа

Какая библиотека используется для работы с файловыми потоками ввода-вывода?

iostream
+fstream
cmath
iomanip

Для генерации потока файлового ввода необходимо создать объект класса ...

+ifstream
ofstream
sstream
iostream

Для генерации потока файлового вывода необходимо создать объект класса ...

ifstream
+ofstream
sstream
iostream

В каком режиме открываются файлы по умолчанию?

+текстовом
бинарном
мультимедийном

С помощью какого оператора можно осуществить чтение из файлового потока?

+>>
<<
&
=

С помощью какого оператора можно осуществить запись в файловый поток?

>>
+<<<
^
=

С помощью какой функции можно закрыть файл?

+close
open
exit
break

С помощью какого флага можно открыть файл в бинарном режиме?

+ios::binary
ios::in
ios::app
ios::out

С помощью какого флага задается режим ввода, при котором данные дописываются в конец файла?

ios::binary
ios::in
+ios::app
ios::out

Какая функция позволяет переместиться в любое место файлового потока вывода?

+seekp
seekg
move
get

Механизм, связывающий воедино код и данные, которыми он манипулирует, а также обеспечивающий их защиту от внешнего вмешательства и неправильного использования, называется...

+инкапсуляция
полиморфизм
наследование

Атрибут, позволяющий с помощью одного интерфейса управлять доступом к целому ряду методов, называется ...

инкапсуляция
+полиморфизм
наследование

Механизм, который позволяет строить иерархии, в которых производные классы получают элементы родительских классов и могут дополнять их или изменять их свойства, называется...

инкапсуляция
полиморфизм
+наследование

Какое ключевое слово используется при описании класса в языке C++?

struct

+class
union
enum

Какое утверждение является ошибочным?

поля класса могут быть любого типа, кроме типа этого же класса
поля класса могут быть описаны с модификатором const
поля класса могут быть описаны с модификатором static
+возможна инициализация полей класса при его описании

Какой режим доступа у полей класса по умолчанию?

public
+private
protected

Какой элемент класса предназначен для инициализации объекта и вызывается автоматически при его создании?

+конструктор
деструктор
дружественная функция
указатель this

Какой элемент класса предназначен для освобождения памяти, занимаемой объектом?

конструктор
+деструктор
дружественная функция
указатель this

Как называется конструктор, который можно вызвать без параметров?

+конструктор по умолчанию
копирующий конструктор
деструктор
закрытый конструктор

Как называется конструктор, который вызывается при инициализации объекта другим объектом этого же класса?

конструктор по умолчанию
+копирующий конструктор
деструктор
закрытый конструктор

```
1 class Mclass
2 {
3     private:
4         int a;
5     public:
6         Mclass();
7         ~Mclass();
8         int b;
9         int get();
10};
```

В какой строке объявлено закрытое поле класса?

+4

6
7
8
9

```
1 class Mclass
2 {
3 private:
4     int a;
5 public:
6     Mclass();
7     ~Mclass();
8     int b;
9     int get();
10};
```

В какой строке объявлено открытое поле класса?

4
6
7
+8
9

```
1 class Mclass
2 {
3 private:
4     int a;
5 public:
6     Mclass();
7     ~Mclass();
8     int b;
9     int get();
10};
```

В какой строке объявлен метод класса?

4
6
7
8
+9

```
1 class Mclass
2 {
3 private:
4     int a;
5 public:
6     Mclass();
7     ~Mclass();
8     int b;
9     int get();
10};
```

В какой строке объявлен конструктор класса?

4
+6
7

8

9

Как обычной функции (не методу класса) предоставить доступ к закрытым полям класса?

объявить ее статической

+объявить ее дружественной

объявит ее константной

любая функция и так имеет доступ к закрытым полям класса

С помощью какого ключевого слова можно заставить компилятор создать только один экземпляр поля класса, который будет использоваться всеми объектами?

const

+static

virtual

friend

Где правильно осуществляется доступ к методам класса?

class student

{

string fio;

int group;

public:

string getFio();

void setGroup(int a);

};

student a;

+cout<<a.getFio();

cout<<getFio(a);

cout<<a->getFio();

cout<<a[1].getFio();

Где правильно осуществляется доступ к методам класса?

class student

{

string fio;

int group;

public:

string getFio();

void setGroup(int a);

};

*student *a;*

cout<<a.getFio();

cout<<getFio(a);

+cout<<a->getFio();

cout<<a[1].getFio();

Что в языке C++ означает ключевое слово *this*?

+указатель на текущий объект класса

указатель на метод класса

спецификатор доступа

С помощью какого ключевого слова объявляются виртуальные методы класса?

const
static
+virtual
friend

При повышающем приведении типов, чтобы вместо метода базового класса вызывался метод производного класса, используют ...

дружественные методы
+виртуальные методы
статические методы

Выберите правильное объявление чисто виртуального метода:

clear virtual f(int);
+virtual f(int)=0;
f(int) = virtual;

Если внутри класса есть виртуальный метод, то класс становится ...

+абстрактным
константным
статическим
дружественным

Выберите несколько правильных вариантов ответа

Какие ограничения накладываются на константные методы класса?

они могут вызываться только для константных объектов
+они могут вызывать только константные методы (50%)
+они не могут менять значение полей класса (50%)
они могут вызываться только внутри конструктора класса

Какие требования выдвигаются при перегрузке функции?

+перегруженные функции должны отличаться типом параметров (50%)
+перегруженные функции должны отличаться количеством параметров (50%)
перегруженные функции должны отличаться типом возвращаемого значения

Какие элементы класса не наследуются?

поля
методы
+конструкторы (33%)
+деструкторы (33%)
+оператор присвоения (33%)

Выберите верные утверждения

виртуальный метод может объявляться с модификатором static
+виртуальный метод не может объявляться с модификатором static (50%)
+виртуальные методы наследуются (50%)
деструктор не может быть виртуальным

Методика проведения контроля

Параметры методики	Значение параметра
Предел длительности всего контроля	10 минут
Последовательность выбора вопросов	Случайная
Предлагаемое количество вопросов	10

Раздел 2. Паттерны проектирование

Контролируемые компетенции ОК 02; ПК 1.2; ПК 1.3; ПК 1.6

Вопросы для устного опроса

1. Назначение порождающих шаблонов проектирования. Перечень порождающих шаблонов.
2. Назначение и структура шаблона «абстрактная фабрика» (abstract factory). Преимущества и недостатки шаблона.
3. Назначение и структура шаблона «строитель» (builder). Преимущества и недостатки шаблона.
4. Назначение и структура шаблона «фабричный метод» (factory method). Преимущества и недостатки шаблона.
5. Назначение и структура шаблона «отложенная (ленивая) инициализация» (lazy initialization). Преимущества и недостатки шаблона.
6. Назначение и структура шаблона «пул одиночек» (multiton). Преимущества и недостатки шаблона.
7. Назначение и структура шаблона «объектный пул» (object pool). Преимущества и недостатки шаблона.
8. Назначение и структура шаблона «прототип» (prototype). Преимущества и недостатки шаблона.
9. Назначение и структура шаблона «инициализация при получении ресурса» (resource acquisition is initialization – RAII). Преимущества и недостатки шаблона.
10. Назначение и структура шаблона «одиночка» (singleton). Преимущества и недостатки шаблона.
11. Назначение структурных шаблонов проектирования. Перечень структурных шаблонов.
12. Назначение и структура шаблона «адаптер» (adapter/ wrapper). Преимущества и недостатки шаблона.
13. Назначение и структура шаблона «мост» (bridge). Преимущества и недостатки шаблона.
14. Назначение и структура шаблона «компоновщик» (composite). Преимущества и недостатки шаблона.
15. Назначение и структура шаблона «декоратор» (decorator). Преимущества и недостатки шаблона.
16. Назначение и структура шаблона «фасад» (facade). Преимущества и недостатки шаблона.
17. Назначение и структура шаблона «приспособленец» (flyweight). Преимущества и недостатки шаблона.
18. Назначение и структура шаблона «заместитель» (proxy). Преимущества и недостатки шаблона.
19. Назначение поведенческих шаблонов проектирования. Перечень поведенческих шаблонов.
20. Назначение и структура шаблона «Цепочка ответственности» (a Chain of responsibility). Преимущества и недостатки шаблона.
21. Назначение и структура шаблона «Команда» (Command). Преимущества и недостатки шаблона.
22. Назначение и структура шаблона «Интерпретатор» (Interpreter). Преимущества и недостатки шаблона.
23. Назначение и структура шаблона «Итератор» (Iterator). Преимущества и недостатки шаблона.

24. Назначение и структура шаблона «Посредник» (Mediator). Преимущества и недостатки шаблона.
25. Назначение и структура шаблона «Хранитель» (Memento). Преимущества и недостатки шаблона.
26. Назначение и структура шаблона «Наблюдатель» (Observer). Преимущества и недостатки шаблона.
27. Назначение и структура шаблона «Состояние» (State). Преимущества и недостатки шаблона.
28. Назначение и структура шаблона «Стратегия» (Strategy). Преимущества и недостатки шаблона.
29. Назначение и структура шаблона «Шаблонный метод» (Template method). Преимущества и недостатки шаблона.
30. Назначение и структура шаблона «Посетитель» (Visitor). Преимущества и недостатки шаблона.

Критерии оценки:

Оценка «отлично» выставляется обучающемуся, который прочно усвоил программный материал в полном объеме, исчерпывающе, грамотно и логически стройно его излагает, четко формулирует основные понятия, приводит соответствующие примеры, уверенно владеет материалом.

Оценка «хорошо» выставляется обучающемуся, который твердо усвоил программный материал, грамотно и по существу излагает его без существенных ошибок, правильно применяет теоретические положения при решении конкретных задач, с небольшими погрешностями приводит формулировки определений, по ходу изложения допускает небольшие пробелы, не искажающие содержания ответа.

Оценка «удовлетворительно» выставляется обучающемуся, который не совсем твердо владеет программным материалом, знает основные теоретические положения изучаемой темы, при ответах допускает малосущественные погрешности, искажения логической последовательности при изложении материала, неточную аргументацию теоретических положений, испытывает затруднения при ответе на дополнительные вопросы.

Оценка «неудовлетворительно» выставляется обучающемуся, имеющему серьезные пробелы в знании учебного материала, допускающему принципиальные ошибки при ответе на вопросы.

Билеты к экзамену

Контролируемые компетенции ОК 02; ПК 1.2; ПК 1.3; ПК 1.6

Вариант 1

В базе данных системы электронного документооборота обрабатываются документы 3-х типов: письма, приказы и распоряжения о командировке. Каждый документ имеет номер, дату и краткую информацию о содержании. Кроме того, в письме указывается тип (входящее/исходящее) и корреспондент, в приказе – подразделение, срок выполнения и ответственный исполнитель, в распоряжении о командировке – сотрудник, период и место назначения.

Необходимо вывести:

1. Полный перечень документов.
2. Содержание выбранного документа (по номеру документа).

Вариант 2

В базе данных компьютерного интернет-магазина хранится номенклатура товаров 3-х типов: материнские платы, процессоры, жесткие диски. Каждый товар имеет номенклатурный номер, наименование и стоимость. Кроме того, для материнских плат указывается тип сокета, количество процессоров, тип оперативной памяти, частота системной шины, для процессоров – тип сокета, количество ядер, тактовая частота, техпроцесс, для жестких дисков – объем, скорость вращения, тип интерфейса.

Необходимо вывести:

1. Полную номенклатуру комплектующих.
2. Детальную информацию по товару (по номенклатурному номеру).

Вариант 3

Информационная система обеспечивает формирование отчета по лабораторным работам студента ИТ-специальности. Рассматриваются дисциплины «Базы данных», «Компьютерные сети», «Программирование». Отчет должен включать следующие разделы: титульный лист, цель работы, задание, теоретические сведения, описание экспериментальной установки, результаты работы, анализ результатов работы, выводы.

Необходимо реализовать ввод данных и подготовку отчета в формате html с учетом особенностей дисциплины (наличие схем и других изображений или текста программы и т.п.).

Вариант 4

Информационная система предназначена для «сборки» комплектации автомобиля. В зависимости от марки и модели комплектация может включать, например, следующие категории элементов: экстерьер, интерьер, комфорт, безопасность, мультимедиа и т.п.

Реализовать «сборку» различных комплектаций для выбранной модели автомобиля и подготовку сравнительной таблицы опций в формате html. При выполнении задания использовать реальный каталог выбранного автопроизводителя.

Вариант 5

В строительной компании информационная система решает задачу оптимального выбора поставщиков строительных материалов (всего 3 поставщика). В строительстве используется 3 вида материалов: кирпич, железобетонные плиты, бетон. Для каждого поставщика задан прайс-лист на материалы и максимальные суточные поставки каждого материала.

Реализовать ввод необходимых данных (включая суточную потребность в материалах) и расчет оптимального плана поставок с использованием методов теории принятия решений. Учесть ситуацию, когда поставщик не поставляет конкретный вид материала.

Вариант 6

Информационная система предназначена для «сборки» моделей автомобилей. Модель автомобиля может включать, например, следующие компоненты: кузов, двигатель, коробка передач, диски/шины (выбрать не менее 3 компонентов). Описать каждый из компонентов классом с подходящим набором атрибутов, реализовать также класс «модель». Реализовать «сборку» различных моделей выбранного автопроизводителя (не менее 3 моделей) на основе реального каталога. Вывести описание моделей в html-файл.

Вариант 7

Реализовать ведение журнала событий программы в форматах: текстовый файл, xml-файл (конкретный формат должен выбираться). Сообщения, посылаемые в журнал, должны иметь тип (информация, предупреждение или ошибка), источник (наименование модуля программы), метку даты/времени и содержание. Реализация должна быть потокобезопасной, т.е. обеспечивать корректную работу журнала в случае, когда в приложении параллельно работает несколько потоков.

Вариант 8

Реализовать менеджер для загрузки файлов с разделяемого сетевого ресурса (файл-сервера) с поддержкой многопоточной загрузки. Пользователь задает URI-адреса нескольких файлов, которые требуется загрузить. Для загрузки используется заданное количество объектов (потоков) – не более 1 потока на 1 файл. Должна поддерживаться авторизация для доступа к файлам.

Вариант 9

В информационной системе хранится организационная структура предприятия (иерархия структурных подразделений). Для каждого структурного подразделения задан код, наименование и перечень должностей по штатному расписанию: наименование должности, количество ставок, оклад.

Необходимо разработать набор классов для представления организационной структуры с использованием заданного шаблона проектирования. Должны быть реализованы возможности вывода штатного расписания для любого подразделения (включая подчиненные подразделения) с расчетом общего количества штатных единиц и суммарного оклада. Программа должна предусматривать добавление и удаление подразделений, должностей.

Вариант 10

Реализовать библиотеку для задания образцов поиска строк (регулярных выражений). Должны поддерживаться:

- поиск строки по точному совпадению;
- поиск строки по неточному совпадению;
- применение оператора AND (и);
- применение оператора OR (или).

Критерии оценки:

Оценка «отлично» выставляется обучающемуся, который правильно выполняет все задания, успешно применяет теоретические знания к решению практических задач с использованием алгоритма как способа автоматизации деятельности.

Оценка «хорошо» выставляется обучающемуся, который: правильно выполняет не менее 80% заданий, применяет теоретические знания к решению практических задач.

Оценка «удовлетворительно» выставляется обучающемуся, который правильно выполняет не менее 60% заданий, не совсем твердо владеет материалом, испытывает затруднения при решении достаточно сложных задач.

Оценка «неудовлетворительно» выставляется обучающемуся, который правильно выполняет менее 60% заданий, имеющему серьезные пробелы в знании учебного материала,

допускающему принципиальные ошибки при выполнении предусмотренных контрольных заданий

Форма промежуточной аттестации по дисциплине экзамен.

Окончательные результаты обучения (формирования компетенций) определяются посредством перевода баллов, набранных студентом в процессе освоения дисциплины, в оценки:

– базовый уровень сформированности компетенции считается достигнутым если результат обучения соответствует оценке «удовлетворительно» (50 до 64 рейтинговых баллов);

– повышенный уровень сформированности компетенции считается достигнутым, если результат обучения соответствует оценкам «хорошо» (65-85 рейтинговых баллов) и «отлично» (86-100 рейтинговых баллов).

Дополнительные контрольные испытания

для студентов, набравших менее 50 баллов (в соответствии с Положением «О модульно-рейтинговой системе»), формируются из числа оценочных средств по темам, которые не освоены студентом.