

Государственное образовательное учреждение
«ПРИДНЕСТРОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
имени Т.Г. Шевченко»

Рыбницкий филиал ПГУ им. Т.Г. Шевченко
Кафедра «Автоматизация технологических процессов и производств»

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

Б.1.В.06 «Объектно-ориентированные программы»

Код

наименование дисциплины

Основной образовательной программы высшего образования по направлению
подготовки 2.15.03.04 «Автоматизация технологических процессов и производств»

индекс

наименование направления

профиль Автоматизация технологических процессов и производств

наименование профиля подготовки

квалификация выпускника бакалавр

форма обучения очная/заочная

год набора: 2019, 2020

Разработчик

преподаватель кафедры АТПиП

Заболотная В.В.

Обсужден на заседании кафедры

« 23 » 09 2021 г.

Протокол № 2

Зав. кафедрой АТПиП, доцент

Федоров В.Е.

Рыбница, 2021 г.

ПАСПОРТ
фонда оценочных средств по учебной дисциплине

Объектно-ориентированные программы
(наименование дисциплины)

1. В результате изучения дисциплины «Объектно-ориентированные программы» обучающийся должен:

1.1 . Знать:

- теоретические основы, методы и способы объектно-ориентированного программирования;
- основные приемы объектно-ориентированного решения задач и способы построения объектно-ориентированных алгоритмов;
- принципы создания современных объектно-ориентированных программных систем.

1.2. Уметь:

- отображать предметную область прикладной задачи в объектную модель программной системы;
- создавать программу на языке объектно-ориентированного программирования на базе построенной объектной модели;
- проектировать диалоговый интерфейс к программе с помощью современных средств разработки приложений.

1.3. Владеть:

- способами проектирования объектно-ориентированных программных систем;
- архитектурой и возможностями современных языков объектно-ориентированного программирования.

2. Программа оценивания контролируемой компетенции:

Текущая аттестация	Контролируемые модули, разделы (темы) дисциплины и их наименование *	Код контролируемой компетенции (или ее части)	Наименование оценочного средства**
1	Раздел 1. Типы и структуры данных	ОК-5, ОПК-3, ПК-1	Комплект тестов
2	Раздел 2. Основы объектно-ориентированного программирования	ОК-5, ОПК-3, ПК-1	Комплект заданий для выполнения контрольной работы Комплект тестов
Промежуточная аттестация		Код контролируемой компетенции (или ее части)	Наименование оценочного средства**
1		ОК-5, ОПК-3, ПК-1	Комплект КИМ

«УТВЕРЖДАЮ»
зав. кафедрой автоматизации
технологических процессов и
производств,
доцент В.Е. Федоров
« 23 » 09 2021 г.

**Вопросы к экзамену
по дисциплине «Объектно-ориентированные программы»
для студентов II, III и IV курсов
направления «Автоматизация технологических процессов и производств»
профиля подготовки
«Автоматизация технологических процессов и производств»,
IV семестр (д/о), VI семестр (з/о)**

1. Переменные и типы переменных в языке программирования ObjectPascal.
2. Среда разработки Delphi: главное окно, окно формы, окно дерева объектов, окно инспектора объектов, окно кода программы.
3. Основы визуального программирования: форма, размещение компонента, реакция на событие.
4. Структура программ Delphi: структура проекта, структура модуля, элементы.
5. Основные этапы разработки программы: разработка алгоритма, кодирование, отладка, тестирование.
6. Основные типы данных (целые и вещественные типы) в Delphi.
7. Работа со строковыми типами данных в Delphi.
8. Символьные типы данных в Delphi.
9. Функции и процедуры для работы со строками и символами.
10. Логический тип данных в Delphi. Логические операции.
11. Управляющая конструкция if. Условный и составной операторы.
12. Свойства компонент. Свойства формы. Свойства кнопки TButton.
13. Инструкция множественного выбора case.
14. Инструкция (цикл с предусловием) WHILE.
15. Концепция объектно-ориентированного программирования.
16. Компоненты Panel, GroupBox, RadioGroup, CheckBox
17. Меню. Свойства Enabled и Visible. Главное и всплывающее меню.
18. Работа с массивами в Delphi. Статические и динамические массивы.
19. Структура проекта в Delphi.
20. Класс TStrings. Свойства TStrings. Методы объекта TStrings.
21. Стандартные диалоги в Delphi.
22. Адрес и имя файла. Функции для работы с файлами в Delphi.
23. Стандартные подпрограммы преобразования типов.
24. Кнопки с изображениями и маскировочное поле ввода.
25. Работа с текстовыми файлами в среде Delphi.

Экзаменатор, ст. преподаватель З.В. Заболотная В.В. Заболотная

«УТВЕРЖДАЮ»
зав. кафедрой, доцент
В.Е. Федоров
« 23 » 09 2021 г.

ПРИДНЕСТРОВСКИЙ УНИВЕРСИТЕТ

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

по дисциплине «Объектно-ориентированные программы»

направление «Автоматизация технологических процессов и производств»

профиль подготовки «Автоматизация технологических процессов и производств»,

II курс, 4 семестр, д/о; III курс, 6 семестр, з/о

1. Среда разработки Delphi: главное окно, окно формы, окно дерева объектов, окно инспектора объектов, окно кода программы.
2. Свойства компонент. Свойства формы. Свойства кнопки TButton.

3. Практическое задание № 1

Экзаменатор ЗВmf В.В. Заболотная
« 23 » 09 2021 г.

«УТВЕРЖДАЮ»
зав. кафедрой, доцент
В.Е. Федоров
« 23 » 09 2021 г.

ПРИДНЕСТРОВСКИЙ УНИВЕРСИТЕТ

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 2

по дисциплине «Объектно-ориентированные программы»

направление «Автоматизация технологических процессов и производств»

профиль подготовки «Автоматизация технологических процессов и производств»,

II курс, 4 семестр, д/о; III курс, 6 семестр, з/о

1. Переменные и типы переменных в языке программирования ObjectPascal.
2. Инструкция множественного выбора case.
3. Практическое задание № 2

Экзаменатор ЗВmf В.В. Заболотная
« 23 » 09 2021 г.

«УТВЕРЖДАЮ»
зав. кафедрой, доцент
В.Е. Федоров
« 23 » 03 2021 г.

ПРИДНЕСТРОВСКИЙ УНИВЕРСИТЕТ

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 3

по дисциплине «Объектно-ориентированные программы»

направление «Автоматизация технологических процессов и производств»

профиль подготовки «Автоматизация технологических процессов и производств»,

II курс, 4 семестр, д/о; III курс, 6 семестр, з/о

1. Инструкция (цикл с предусловием) WHILE

2. Основы визуального программирования: форма, размещение компонента, реакция на событие.

3. Практическое задание № 3

Экзаменатор З.В. Заболотная В.В. Заболотная
« 23 » 03 2021 г.

«УТВЕРЖДАЮ»
зав. кафедрой, доцент
В.Е. Федоров
« 23 » 03 2021 г.

ПРИДНЕСТРОВСКИЙ УНИВЕРСИТЕТ

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 4

по дисциплине «Объектно-ориентированные программы»

направление «Автоматизация технологических процессов и производств»

профиль подготовки «Автоматизация технологических процессов и производств»,

II курс, 4 семестр, д/о; III курс, 6 семестр, з/о

1. Структура программ Delphi: структура проекта, структура модуля, элементы.

2. Концепция объектно-ориентированного программирования

3. Практическое задание № 4

Экзаменатор З.В. Заболотная В.В. Заболотная
« 23 » 03 2021 г.

«УТВЕРЖДАЮ»
зав. кафедрой, доцент
В.Е. Федоров
« 23 » 09 2021 г.

ПРИДНЕСТРОВСКИЙ УНИВЕРСИТЕТ

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 5

по дисциплине «Объектно-ориентированные программы»

направление «Автоматизация технологических процессов и производств»

профиль подготовки «Автоматизация технологических процессов и производств»,

II курс, 4 семестр, д/о; III курс, 6 семестр, з/о

1. Концепция объектно-ориентированного программирования.

2. Основные этапы разработки программы: разработка алгоритма,
кодирование, отладка, тестирование.

3. Практическое задание № 5

Экзаменатор З.В. Заболотная В.В. Заболотная
« 23 » 09 2021 г.

«УТВЕРЖДАЮ»
зав. кафедрой, доцент
В.Е. Федоров
« 23 » 09 2021 г.

ПРИДНЕСТРОВСКИЙ УНИВЕРСИТЕТ

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 6

по дисциплине «Объектно-ориентированные программы»

направление «Автоматизация технологических процессов и производств»

профиль подготовки «Автоматизация технологических процессов и производств»,

II курс, 4 семестр, д/о; III курс, 6 семестр, з/о

1. Основные этапы разработки программы: разработка алгоритма,
кодирование, отладка, тестирование.

2. Оператор цикла с предусловием while

3. Практическое задание № 6

Экзаменатор З.В. Заболотная В.В. Заболотная
« 23 » 09 2021 г.

«УТВЕРЖДАЮ»
зав. кафедрой, доцент
В.Е. Федоров
« 23 » 09 2021 г.

ПРИДНЕСТРОВСКИЙ УНИВЕРСИТЕТ

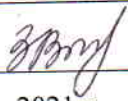
ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 7

по дисциплине «Объектно-ориентированные программы»
направление «Автоматизация технологических процессов и производств»
профиль подготовки «Автоматизация технологических процессов и производств»,
II курс, 4 семестр, д/о; III курс, 6 семестр, з/о

1. Основные типы данных (целые и вещественные типы) в Delphi.

2. Концепция объектно-ориентированного программирования

3. Практическое задание № 7

Экзаменатор  В.В. Заболотная
« 23 » 09 2021 г.

«УТВЕРЖДАЮ»
зав. кафедрой, доцент
В.Е. Федоров
« 23 » 09 2021 г.

ПРИДНЕСТРОВСКИЙ УНИВЕРСИТЕТ

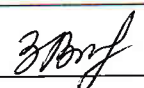
ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 8

по дисциплине «Объектно-ориентированные программы»
направление «Автоматизация технологических процессов и производств»
профиль подготовки «Автоматизация технологических процессов и производств»,
II курс, 4 семестр, д/о; III курс, 6 семестр, з/о

1. Работа со строковыми типами данных в Delphi.

2. Компоненты Panel, GroupBox, RadioGroup, CheckBox

3. Практическое задание № 8

Экзаменатор  В.В. Заболотная
« 23 » 09 2021 г.

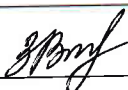
«УТВЕРЖДАЮ»
зав. кафедрой, доцент
В.Е. Федоров
« 23 » 01 2021 г.

ПРИДНЕСТРОВСКИЙ УНИВЕРСИТЕТ

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 9

по дисциплине «Объектно-ориентированные программы»
направление «Автоматизация технологических процессов и производств»
профиль подготовки «Автоматизация технологических процессов и производств»,
II курс, 4 семестр, д/о; III курс, 6 семестр, з/о

1. Символьные типы данных в Delphi.
2. Меню. Свойства Enabled и Visible. Главное и всплывающее меню.
3. Практическое задание № 9

Экзаменатор  В.В. Заболотная
« 23 » 01 2021 г.

«УТВЕРЖДАЮ»
зав. кафедрой, доцент
В.Е. Федоров
« 23 » 01 2021 г.

ПРИДНЕСТРОВСКИЙ УНИВЕРСИТЕТ

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 10

по дисциплине «Объектно-ориентированные программы»
направление «Автоматизация технологических процессов и производств»
профиль подготовки «Автоматизация технологических процессов и производств»,
II курс, 4 семестр, д/о; III курс, 6 семестр, з/о

1. Функции и процедуры для работы со строками и символами
2. Структура проекта в среде Delphi.
3. Практическое задание № 10

Экзаменатор  В.В. Заболотная
« 23 » 01 2021 г.

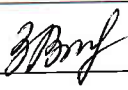
«УТВЕРЖДАЮ»
зав. кафедрой, доцент
В.Е. Федоров
«23» 09 2021 г.

ПРИДНЕСТРОВСКИЙ УНИВЕРСИТЕТ

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 11

по дисциплине «Объектно-ориентированные программы»
направление «Автоматизация технологических процессов и производств»
профиль подготовки «Автоматизация технологических процессов и производств»,
II курс, 4 семестр, д/о; III курс, 6 семестр, з/о

1. Логический тип данных в Delphi. Логические операции.
2. Работа с массивами в Delphi. Статические и динамические массивы
3. Практическое задание № 11

Экзаменатор  В.В. Заболотная
«23» 09 2021 г.

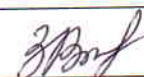
«УТВЕРЖДАЮ»
зав. кафедрой, доцент
В.Е. Федоров
«23» 09 2021 г.

ПРИДНЕСТРОВСКИЙ УНИВЕРСИТЕТ

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 12

по дисциплине «Объектно-ориентированные программы»
направление «Автоматизация технологических процессов и производств»
профиль подготовки «Автоматизация технологических процессов и производств»,
II курс, 4 семестр, д/о; III курс, 6 семестр, з/о

1. Управляющая конструкция if. Условный и составной операторы.
2. Класс TString. Свойства TString. Методы объекта TString.
3. Практическое задание № 12

Экзаменатор  В.В. Заболотная
«23» 09 2021 г.

«УТВЕРЖДАЮ»
зав. кафедрой, доцент
В.Е. Федоров
«23» 09 2021 г.

ПРИДНЕСТРОВСКИЙ УНИВЕРСИТЕТ

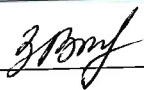
ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 13

по дисциплине «Объектно-ориентированные программы»
направление «Автоматизация технологических процессов и производств»
профиль подготовки «Автоматизация технологических процессов и производств»,
II курс, 4 семестр, д/о; III курс, 6 семестр, з/о

1. Стандартные диалоги в Delphi.

2. Кнопки с изображениями и маскировочное поле ввода.

3. Практическое задание № 13

Экзаменатор  В.В. Заболотная
«23» 09 2021 г.

«УТВЕРЖДАЮ»
зав. кафедрой, доцент
В.Е. Федоров
«23» 09 2021 г.

ПРИДНЕСТРОВСКИЙ УНИВЕРСИТЕТ

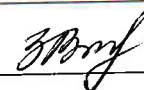
ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 14

по дисциплине «Объектно-ориентированные программы»
направление «Автоматизация технологических процессов и производств»
профиль подготовки «Автоматизация технологических процессов и производств»,
II курс, 4 семестр, д/о; III курс, 6 семестр, з/о

1. Адрес и имя файла. Функции для работы с файла в Delphi.

2. Вывод сообщений на форму. Метод ShowMessage.

3. Практическое задание № 14

Экзаменатор  В.В. Заболотная
«23» 09 2021 г.

«УТВЕРЖДАЮ»
зав. кафедрой, доцент
В.Е. Федоров
« 23 » 09 2021 г.

ПРИДНЕСТРОВСКИЙ УНИВЕРСИТЕТ

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 15

по дисциплине «Объектно-ориентированные программы»

направление «Автоматизация технологических процессов и производств»

профиль подготовки «Автоматизация технологических процессов и производств»,

II курс, 4 семестр, д/о; III курс, 6 семестр, з/о

1. Стандартные подпрограммы преобразования типов.

2. Работа с текстовыми файлами в среде Delphi.

3. Практическое задание № 15

Экзаменатор ЗВmf В.В. Заболотная
« 23 » 09 2021 г.

«УТВЕРЖДАЮ»
зав. кафедрой, доцент
В.Е. Федоров
« 23 » 09 2021 г.

ПРИДНЕСТРОВСКИЙ УНИВЕРСИТЕТ

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 16

по дисциплине «Объектно-ориентированные программы»

направление «Автоматизация технологических процессов и производств»

профиль подготовки «Автоматизация технологических процессов и производств»,

II курс, 4 семестр, д/о; III курс, 6 семестр, з/о

1. Среда разработки Delphi: основные понятия.

2. Управляющая конструкция if. Условный и составной операторы.

3. Практическое задание № 16

Экзаменатор ЗВmf В.В. Заболотная
« 23 » 09 2021 г.

Государственное образовательное учреждение
«ПРИДНЕСТРОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
имени Т.Г. Шевченко»
Рыбницкий филиал ПГУ им. Т.Г. Шевченко
Кафедра «Автоматизация технологических процессов и производств»

Тест

по дисциплине Объектно-ориентированные программы
(наименование дисциплины)

Вопрос № 1

Какой язык программирования представляет собою реализацию концепции процедурного программирования?

1. COBOL
2. SQL
3. FORTRAN
4. JAVA

Вопрос № 2

Какой язык программирования представляет собою реализацию концепции структурного программирования?

1. C++
2. MODULA
3. JAVASCRIPT
4. PROLOG

Вопрос № 3

Не имеет отношения к описанию степени доступа к элементам класса слово ...

1. common
2. public
3. private
4. protected

Вопрос № 4

Ранжированная, или упорядоченная система абстракций - это ...

1. модульность
2. система ограничений
3. иерархия
4. полиморфизм

Вопрос № 5

Какой из языков программирования поддерживает программирование для платформы .NET ?

1. C#
2. C++

3. C
4. Java

Вопрос № 6

В Delphi 2005 появилось новое средство оптимизации кода - inlining. Какое из приводимых ниже определений ему соответствует ?

1. Подстановка компилятором кода метода в точку обращения к методу
2. Передача компилятором управления в новую точку программы
3. Настройка компилятором кадра стека перед обращением к подпрограмме
4. Передача компилятором управления по адресу расположения подпрограммы

Вопрос № 7

К какому классу языков программирования следует отнести язык C++ ?

1. Язык процедурного типа
2. Язык декларативно-функционального типа
3. Язык объектно-ориентированного типа
4. Язык сценарного типа

Вопрос № 8

Процесс вычленения ряда характеристик некоторого объекта предметной области, позволяющий отличить его от всех других видов это - ...

1. иерархия
2. модульность
3. ограничение доступа
4. абстрагирование

Вопрос № 9

Какие термины относятся к объектно-ориентированной парадигме программирования ?

1. Production system, rule based system
2. Production, function, rule
3. Object, property, event
4. Structure, algorithm, automatic

Вопрос № 10

Концептуальная основа ООП базируется на 4-х важнейших элементах

1. Абстрагирование, ограничение доступа, модульность, защита
2. Абстрагирование, ограничение доступа, защита, иерархия
3. Защита, ограничение доступа, модульность, иерархия
4. Абстрагирование, ограничение доступа, модульность, иерархия

Вопрос № 11

Поля, процедуры и функции, используемые только внутри данного класса, находятся в разделе ...

1. published
2. public

3. protected
4. private

Вопрос № 12

Открытые свойства, которые доступны в процессе проектирования на странице Object Inspector, объявляются в разделе ...

1. private
2. published
3. protected
4. public

Вопрос № 13

Объявления элементов, доступных для внешнего использования, находятся в разделе ...

1. private
2. protected
3. initialization
4. public

Вопрос № 14

Какой из приведённых методов не обеспечивает удаления формы ... ?

1. Free
2. Destroy
3. Release
4. Hide

Вопрос № 15

Какой компонент из перечисленных является визуальным ...

1. Splitter
2. ColorDialog
3. FontDialog
4. SavePictureDialog

Вопрос № 16

Объявления методов приводятся в разделе описания ...

1. interface
2. uses
3. implementation
4. var

Вопрос № 17

Реализации методов приводятся в разделе описания ...

1. interface

2. uses
3. implementation
4. type

Вопрос № 18

Метод, объявленный в базовом классе как виртуальный, в дальнейшем во всех классах-наследниках ...

1. становится статическим
2. становится динамическим
3. становится реальным
4. остаётся виртуальным

Вопрос № 19

Наличие конструктора и деструктора обязательно при использовании ... методов

1. статических
2. динамических
3. виртуальных
4. подстановочных

Вопрос № 20

Конструктор класса - это метод, который вызывается при создании экземпляра класса с целью ...

1. загрузки всех методов класса в память
2. задания свойств объекта
3. выделения памяти под создаваемый экземпляр класса и инициализации этой памяти
4. выгрузки методов класса из памяти

Вопрос № 21

Когда компилятор встраивает код функции вместо её вызова, то функцию называют...

1. Деструктором
2. Конструктором
3. Обратной
4. Подставляемой

Вопрос № 22

Из событий, связанных с формой, один раз возникает такое событие, как ...

1. OnPaint
2. OnResize
3. OnActivate
4. OnCreate

Вопрос № 23

И статический, и динамический методы могут быть перегружены при использовании в их описании ключевого слова ...

1. override
2. abstract
3. overload
4. reintroduce

Вопрос № 24

Закрытие главной формы проекта приводит к ...

1. завершению работы приложения
2. запросу на создание другой формы проекта
3. созданию другой формы проекта
4. показу следующей открытой формы

Вопрос № 25

Кто является автором (создателем) языка программирования Pascal ?

1. Б. Паскаль
2. Г. Буч
3. Н. Вирт
4. Б. Страуструп

Вопрос № 26

Когда был создан язык программирования Pascal ?

1. В 1940-х годах
2. В 1950-х годах
3. В 1960-х годах
4. В 1970-х годах

Вопрос № 27

Если виртуальный метод перегружен в классе-наследнике, при обращении к этому методу будет вызван вариант метода, соответствующий ...

1. имени объекта, указанному при вызове
2. имени экземпляра класса-наследника
3. имени экземпляра класса-родителя
4. имени экземпляра абстрактного класса

Вопрос № 28

Какая из программных конструкций не используется в языке программирования Pascal ?

1. FOR ... NEXT ...
2. GOTO ...
3. :=
4. FOR ... IN

Вопрос № 29

Какая из программных конструкций не используется в языке программирования Delphi ?

1. LOOP ...
2. FOR ... NEXT ...
3. WHILE ... DO ...
4. REPEAT ... UNTIL ...

Критерии оценки*:

- оценка «отлично» выставляется студенту, если он набрал 25 – 29 баллов;
- оценка «хорошо» - 20 – 24 баллов;
- оценка «удовлетворительно» - 16 – 19 баллов;
- оценка «неудовлетворительно» менее 16 баллов.

* За каждый правильный ответ на тестовое задание выставляется 1 балл.

Ст. преподаватель


(подпись)

Заболотная В.В.
(ФИО)

« 23 » 09

2021 г.

Ответы к тестовым заданиям

№ вопроса	Тест
1.	3
2.	2
3.	1
4.	3
5.	1
6.	1
7.	3
8.	4
9.	3
10.	4
11.	4
12.	2
13.	4
14.	4
15.	1
16.	1
17.	3
18.	4
19.	2
20.	3
21.	4
22.	4
23.	3
24.	1
25.	3
26.	3
27.	1
28.	4
29.	1

Государственное образовательное учреждение
«ПРИДНЕСТРОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
имени Т.Г. Шевченко»

Комплект заданий для контрольной работы

по дисциплине **Объектно-ориентированные программы**
(наименование дисциплины)

1. Реализовать консольную версию карточной игры блекджек (или очко).

Правила можно реализовать любые на ваше усмотрение, хоть самые простые: от двойки до десятки = от 2 до 10 очков, туз = 10 или 1 очко, любая "картинка" = 10 очков, игра до 21 очка игрок против дилера, дилер берет карты последним, при равенстве очков - "ничья", без сплитов, даблов и прочих дополнительных опций. Вероятности выпадения карт одинаковые и не меняются со временем ("бесконечная" колода). Правила должны быть реальными - в том смысле, что по ним действительно имеет смысл играть.

Реализовать возможность игры как "человек против компьютера", так и "человек против человека" и "компьютер против компьютера" (тут как раз появится полиморфизм).

На правах подсказки: классы, которые (вероятно) могут вам понадобиться (но окончательное решение за вами!): карта; игрок и два его наследника: компьютер и человек; игра; вспомогательные классы для вывода информации на экран.

На интерес: сделать возможность столкнуть компьютер против компьютера 1000000 раз с целью получения статистической оценки эффективности различных стратегий игры.

2. Реализовать набор классов для работы с матрицами и векторами.

Вектор можно считать разновидностью матрицы. Предусмотреть операции сложения, вычитания и умножения, конструкторы нулевой, единичной, диагональной и др. матриц. Реализовать два способа хранения: плотная матрица и разреженная матрица (см., например, тут, также см. тут).

В целях тестирования сделать консольную программу, которая позволяет записывать матрицы в "переменные" (т.е. давать им имена) и тестировать все реализованные операции.

3. В программе для управления оценками студентов по ООП требуется реализовать подсчет статистики успеваемости всего курса по итогам семестра.

Для этого вы разрабатываете специальный класс "агрегатор оценок". "На вход" ему подают оценки отдельных студентов по 100-балльной шкале (1 студент - 1 оценка), а "на выходе" - получают следующую информацию: общее количество оценок; количество А-шек, В-шек и т.д. средний балл; количество и средний балл тех, кто "сдал" (т.е. имеет оценку Е или выше).

Кроме того, иногда агрегации необходимо подвергнуть оценки отдельных студентов отдельной группы, а иногда - из уже агрегированных оценок группы посчитать статистику по всему курсу (по всем группам). Также нередко возникает необходимость к уже посчитанной статистике по всем группам добавить оценки отдельных студентов.

Также у вас нет уверенности, что применяемая система оценивания всегда будет оставаться постоянной - может быть, 100-балльную систему сменит 5-балльная, а вместо оценок А,В,С,Д станут применяться А++, А+, А, В++, В+, В, С++, С+, С - причем с другой разбивкой по диапазонам баллов. Нужно предусмотреть легкую замену системы оценивания.

На правах подсказки: классы, которые (вероятно) могут вам понадобиться (но окончательное решение за вами!): собственно, агрегатор (который умеет агрегировать как отдельные оценки - так и другие агрегаторы); "система оценивания", которую можно "подключать" к агрегатору; классы для реализации консольного интерфейса для

тестирования того, что получилось. Замечание: от агрегатора не требуется хранить сами оценки!

4. Разработайте набор удобных классов для реализации "иерархического голосового меню" (как в "инфоцентре" вашего мобильного оператора).

Разумеется, реализовывать синтез голоса от вас не требуется - достаточно вывести текст на экран (будем считать, что этот текст впоследствии направляется на "внешний" синтезатор голоса).

Разработанный набор классов должен обеспечивать следующие возможности: вывод меню и чтение пользовательского выбора; "навешивание" на пункты меню нового выбора - подменю (а также меню 3-го уровня и так далее); операцию "вывод текстовой информации" (для "навешивания" на пункт меню); операцию "проверить баланс" (для "навешивания" на пункт меню); операцию "пополнить счет" (проблему взаимодействия с банком можно оставить в стороне) (для "навешивания" на пункт меню); еще пару операций с абонентским аккаунтом (на ваш выбор).

Таким образом, пользуясь вашими классами, программисты на телефонной станции смогут согласно своим корпоративным требованиям построить меню для своих абонентов.

Для целей тестирования, постройте из этих классов свой небольшой (но похожий на реальный! - можно взять реальную структуру меню своего оператора) пример меню, демонстрирующий возможности всех ваших классов. Не забудьте "соединение с оператором"! ;)

На правах подсказки: классы, которые (вероятно) могут вам понадобиться (но окончательное решение за вами!): абстрактный "пункт меню"; подменю (является "пунктом меню"); пункт меню "вывод текста", "пополнения счета" и т.п.; классы для реализации текстового взаимодействия с пользователем.

Продумайте, как программисты телефонной компании смогут добавлять к вашему набору классов "нестандартные" (т.е. не предусмотренные вами заранее) операции - вроде соединения с оператором или перечисления денег зоопарку.

5. Разработать игру в крестики-нолики 3x3 для игры против человека или компьютера с консольным интерфейсом.

Реализовать возможность игры как "человек против компьютера", так и "человек против человека" и "компьютер против компьютера" (тут как раз появится полиморфизм).

На правах подсказки: классы, которые (вероятно) могут вам понадобиться (но окончательное решение за вами!): поле; игрок (и две различные его реализации); классы для реализации текстового взаимодействия с пользователем. Ваш ИИ должен играть не как попало, а хорошо!

Не забудьте, что ваши классы должны быть пригодны для использования в том числе в среде, в которой вообще отсутствуют System.in и System.out! - должно быть некоторое абстрагирование от конкретного формата взаимодействия с пользователем!

6. Разработать простой набор классов для демонстрации маршрутизации в сетях.

Сеть состоит из "узлов". Каждый узел может быть либо конечным (компьютер), либо промежуточным (маршрутизатор). Каждый узел имеет адрес (целое число) и название (для удобства). При желании названия можно использовать вместо адресов. Каждый промежуточный может быть соединен двунаправленными каналами "точка-точка" или несколькими другими узлами. В пределах каждого узла эти каналы имеют локальную нумерацию 0,1,2 и т.д.

Конечные узлы подключаются ровно одним каналом ровно к одному из промежуточных узлов. Каждый конечный узел может быть как инициатором отправки сообщения, так и его получателем. Промежуточные узлы используются только для переправки чужих сообщений. На каждом промежуточном узле имеется таблица маршрутизации с двумя столбцами: "адрес назначения", "локальный номер канала для отправки".

Промежуточные узлы переправляют все полученные сообщения по тому каналу, который записан в его их таблице маршрутизации в ячейке, соответствующей адресу конечного получателя сообщения. Конечные узлы все сообщения отправляют по своему единственному каналу. Каждое сообщение содержит (как минимум) три поля: адрес отправителя, адрес получателя, содержимое (обычно - текстовая строка).

Вам требуется разработать набор классов, из которых "в main'e" можно собрать сеть и настроить в ней таблицы маршрутизации. После чего отправить между некоторой парой конечных узлов сообщение - и увидеть на экране, как оно "двигалось" по сети (пример):

7. Разработать набор классов для перевода единиц измерения:

градусы Цельсия в градусы Фаренгейта и обратно;

фунты в килограммы и обратно;

время в часах, минутах и секундах (три числа) в секунды (одно целое число), минуты (одно дробное) или часы (одно дробное) и обратно;

градусы, минуты и секунды (три числа) в градусы (одно дробное) и обратно;

футы и дюймы (два числа) в метры и обратно.

В этих же классах реализовать разбор текстовых строк с обозначениями составных величин: 05:28:14 (5 часов, 28 мин, 14 сек), 50 14' 47" (50 градусов, 14 мин, 47 сек), 4' 08" (4 фута и 8 дюймов).

Пользователю предложить меню с выбором, что во что конвертировать. Предусмотреть легкое (с точки зрения программиста) добавление в систему новых вариантов перевода величин.

8. Разработать упрощенный набор классов для работы с расписанием занятий на факультете.

Предусмотреть наличие "мигалок" а также проведение "общих" лекций - несколько групп в одной аудитории. Проверку непротиворечивости расписания (один и тот же преподаватель одновременно в разных местах, занятия по двум разным предметам одновременно в одной аудитории) делать не требуется. В расписании хранятся предметы, группы, преподаватели, дни недели, номера пар и аудиторий (разделение лекция/практика можно игнорировать).

На правах подсказки: классы, которые (вероятно) могут вам понадобиться (но окончательное решение за вами!): расписание; преподаватель; ячейка расписания и три ее разновидности: пара с одной группой, "мигалка", пара с несколькими группами; "день" расписания; классы для реализации текстового взаимодействия с пользователем.

Рекомендация: расписание можно хранить как двумерный массив ДНИxГРУППЫ или как трехмерный ДНИxПАРЫxГРУППЫ.

В тестирующей программе предусмотреть просмотр расписания по всем группам в заданный день и по всем дням для заданной группы, редактирование ячеек расписания. Сразу после запуска программа уже заполнена реальным расписанием 1 и 2 курса на пн, вт и ср.

9. Разработать набор классов для конструирования тестов (для проверки знаний).

Тест - это упорядоченный набор вопросов. Вопросы бывают с единственным выбором, множественным выбором и со свободным ответом (вводится число или слово).

На правах подсказки: классы, которые (вероятно) могут вам понадобиться (но окончательное решение за вами!): абстрактный вопрос и 3 его разновидности; тест (как набор вопросов); результат теста (как набор ответов) с вычислением оценки; классы для реализации текстового взаимодействия с пользователем.

В целях демонстрации "защиты" в программу какой-либо реальный (можно со своими дополнениями) тест - чтобы его можно было пройти и увидеть свою оценку, а также посмотреть, какие из них были неправильными и каков правильный ответ на соответствующий вопрос.

10. Разработать три класса – один базовый (может быть абстрактным классом или интерфейсом) и два производных от него – согласно своему варианту задания. Используя свой класс Vector из л.р.№3, адаптированный для хранения объектов этих классов, реализовать «программу-каталог» со следующими возможностями:

Просмотр всего содержимого каталога.

Добавление нового элемента в заданную позицию (со сдвигом остальных) – в том числе в конец.

Удаление элемента в заданной позиции (со сдвигом остальных).

Редактирование элемента в заданной позиции.

Несколько вариантов разумного – на Ваш взгляд – поиска по каталогу (результат поиска – не всегда единственный элемент!).

Требования и рекомендации

На хорошую оценку ваша программа должна уметь искать по одному "общему" критерию (который применим к обоим подклассам) и одному специфичному критерию - который применим только к одному из подклассов.

Каталог должен быть смешанным: одновременно в нем могут храниться объекты обоих производных классов (совет: для этого обычно хранят в векторе ссылки на базовый класс (или интерфейс)).

Полиморфное поведение реализуется посредством переопределения (override) методов.

Ввод с клавиатуры, вывод на экран, редактирование объекта – безусловно, являются операциями над объектами класса (т.е., скорее всего, станут его методами) (но если вас смущает потеря таким классом универсальности - т.е. его будет неудобно использовать в Android-приложении - напишите к каждому предметному классу по одному "классу ввода-вывода" в духе "ConsoleIOBus, ConsoleIOTruck" и т.п.).

Критерии оценки:

- оценка «отлично» выставляется студенту, если он выполнил все задания правильно;

- оценка «хорошо» - выполнил все задания, иногда ошибался;

- оценка «удовлетворительно» - часто ошибался, выполнил правильно только половину заданий;

- оценка «неудовлетворительно» почти ничего не смог выполнить правильно

ст. преподаватель


(подпись)

В.В. Заболотная
(ФИО)

« 23 » 09

2021 г