Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Ростовский государственный экономический университет (РИНХ)»
Таганрогский институт имени А. П. Чехова (филиал) «РГЭУ (РИНХ)»

УТВЕРЖДАЮ
Директор филиала
_____ А.Ю. Голобородько
«___» _____ 2016г.

Рабочая программа дисциплины

Б1.В.ДВ.6.2 ОСНОВЫ ПАРАЛЛЕЛЬНОГО ПРОГРАММИРОВАНИЯ

Направление подготовки

09.03.03 «Прикладная информатика»

Профиль

09.03.03.01 «Прикладная информатика в менеджменте»

Уровень образования

бакалавриат

Таганрог
2016 г.

ФАКУЛЬТЕТ	1	физики, математики, информатики
КАФЕДРА	2	информатики
	(код)	(наименование)

ОБЩИЙ ОБЪЕМ* работы обучающихся в час.	уч. план **108**	Очная форма	Заочная форма 4г 6 м	
Всего аудиторных занятий, час., в том числе:			**12**	
- лекций,			4	
по семестрам			4 (7)	
- лабораторные занятия,				
по семестрам				
- практические занятия,			8	
по семестрам			4 (7)	4 (8)
В интерактивной форме, час			2	
Всего самостоятельной работы, час., в том числе:			**92**	
- контрольные работы по семестрам				
- курсовые работы по семестрам				
- курсовые проекты по семестрам				
- другое			28 (7)	64 (8)
Изучено и переаттестовано, час.				
Зачеты, по семестрам, час				4 (8)
Экзамены, по семестрам, час				
Всего ЗЕТ по учебному плану		3		

* Объем часов по всем видам работ переносится из учебного плана.

ОСНОВАНИЕ

ФГОС ВО по направлению подготовки направление 09.03.03 «Прикладная информатика» (квалификация «бакалавр») утвержден Приказом Министерства образования и науки Российской Федерации от 12.03.2015 г. №207

Учебный план направления 09.03.03 «Прикладная информатика»профиль 09.03.03.01 «Прикладная информатика в менеджменте» одобрен Ученым советом вуза 31.05.2016 г. протокол № 11.

АВТОР (Ы) канд. техн. наук, доцент кафедры математики		Драгныш Н.В.	02.06.2016
(ученая степень, звание, должность)	(подпись)	(Ф.И.О.)	(дата)

ОБСУЖДАЛАСЬ И СОГЛАСОВАНА

Кафедрой информатики		Ромм Я.Е.	02.06.2016
	(подпись)	(Ф.И.О.)	(дата)
Факультет физики, математики, информатики		Донских С.А.	09.06.2016
(наименование)	(подпись)	(Ф.И.О.)	(дата)

1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

1.1. **Цели** освоения дисциплины: формирование системы знаний, умений и навыков, связанных с особенностями параллельных вычислений и современных технологий программирования как базы для развития универсальных компетенций и основы для развития профессиональных компетенций.

1.2. **Задачи:**

- формирование у выпускника знаний, способствующих решению профессиональных задач с помощью современных технологий программирования и организации вычислений;
- исследование методов и средств параллельного программирования при решении практических задач;
- повышение уровня грамотности в области параллельного программирования и параллельных вычислительных систем;
- формирование профессиональной информационной культуры;
- стимулирование самостоятельной деятельности по освоению содержания дисциплины и формированию необходимых компетенций.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

2.1. **Цикл (раздел) ОП:** Б1.В.

2.2. **Связь с другими дисциплинами учебного плана**

Перечень предшествующих дисциплин	Перечень последующих дисциплин, видов работ
информатика и программирование вычислительные системы, сети и телекоммуникации	ВКР

3. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Вариант 3.

Студент должен знать: *основные методы, алгоритмы и подходы параллельного программирования, основные инструментальные средства и решения для параллельного программирования, основные информационные ресурсы и источники в области параллельного программирования (ОПК-4, ПК-22).*

Студент должен уметь: *программировать программы использующие параллельные вычисления, анализировать рынок программно-технических средств, информационных продуктов и услуг в области параллельного программирования, выбирать необходимые для организации информационные ресурсы и источники знаний в электронной среде (ОПК-4, ПК-22).*

Студент должен владеть: *навыками написания и отладки параллельных программ при решении практических задач, навыками поиска прикладных решений при решении практических задач параллельного программирования, навыками работы с современными информационными ресурсами и источниками знаний в области параллельного программирования (ОПК-4, ПК-22).*

У студента должны быть сформированы элементы следующих компетенций:

ОПК-4: способностью решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.

ПК-22: способностью анализировать рынок программно-технических средств, информационных продуктов и услуг для создания и модификации информационных систем.

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1. Аудиторные занятия − заочная форма обучения

Неделя	Кол. час	в том числе в интерактивной форме, час.	Вид занятия, модуль, тема и краткое содержание	Формируемые компетенции
	4		**Лекции**	
	4		**Модуль 1 Параллельные вычисления и приложения**	ОПК-4 ПК-22
	4		Архитектура ВС. Классификация вычислительных систем. Пути достижения параллелизма. Параллелизм на уровне команд, потоков, приложений. Анализ эффективности параллельных вычислений. Закон Амдала. Основные этапы разработки параллельных приложений: декомпозиция, выявление информационных зависимостей между подзадачами, масштабирование подзадач и балансировка нагрузки для каждого процессора.	ОПК-4 ПК-22
	8	***2***	**Практические занятия /семинары**	
	8	***2***	**Модуль 2 Параллельное программирование на C#**	ОПК-4 ПК-22
	4	*1*	Клеточная модель "Игра Жизнь" Дж.Конвея	ОПК-4 ПК-22
	4	*1*	Знакомство с "Визуализатором параллелизма" в Visual Studio	ОПК-4 ПК-22
-	**-**	**-**	**Лабораторные занятия**	

4.2. Самостоятельная работа студента – заочная форма обучения

Неделя	Кол. час	Темы, разделы, вынесенные на самостоятельную подготовку, тематика рефератной работы, контрольных работ, рекомендации по использованию литературы, ЭВМ и др.	Формируемые компетенции
	52	**Темы, разделы, вынесенные на самостоятельную подготовку**	ОПК-4 ПК-22
	52	**Модуль 2 Параллельное программирование на C#** Работа с потоками. Многопоточная обработка. Средства синхронизации. Конкурентные коллекции. Работа с задачами.	ОПК-4 ПК-22

		Шаблоны параллелизма Parallel. Технология PLINQ. Планировщик задач. Типовые модели параллельных приложений.	
	40	**Темы и вопросы, определяемые преподавателем с учетом интересов студента**	ОПК-4 ПК-22
	92	**Общая трудоемкость самостоятельной работы (час)**	
	4	**Подготовка к зачету**	ОПК-4 ПК-22

5. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

5.1. Фонд оценочных средств для проведения промежуточной аттестации

Вопросы для подготовки к зачету:

1. Параллельные вычисления. Мотивы параллелизма. Классификация вычислительных систем.
2. Архитектура однопроцессорной машины. Мультикомпьютеры с распределенной памятью. Мультипроцессор с разделяемой памятью.
3. Режимы выполнения независимых частей программы. Уровни параллелизма в многоядерных архитектурах.
4. Анализ эффективности параллельных вычислений. Пределы параллелизма.
5. Проблемы разработки параллельных приложений. Декомпозиция. Декомпозиция по данным. Масштабирование подзадач.
6. Проблемы разработки параллельных приложений. Проблема гонки данных. Проблемы синхронизации.
7. Проблемы разработки параллельных приложений. Проблемы кэшируемой памяти. Модели параллельных приложений
8. Работа с потоками. Структура потока. Состояния потоков. Переключение контекста. Работа с потоками в C#
9. Средства синхронизации. Средства для взаимного исключения. Monitor. Mutex.
10. Средства синхронизации. Сигнальные сообщения. Семафоры. Атомарные операторы.
11. Конкурентные коллекции. Эффективность конкурентных коллекций
12. Работа с задачами. Статусы задачи. Работа с данными в задаче.
13. Шаблоны параллелизма Parallel. Параллельные циклы. Параллельный запуск.
14. Технология PLINQ. Эффективность распараллеливания.
15. Планировщик задач. Опция PreferFairness. Стратегии.
16. Типовые модели параллельных приложений. Модель делегирования.
17. Типовые модели параллельных приложений. Модель с равноправными узлами. Модель конвейера

5.2. Фонд оценочных средств для проведения текущего контроля

Структура и содержание фонда оценочных средств представлены в Приложении 1 к рабочей программе дисциплины

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1. Основная и дополнительная литература

№	Выходные данные	Количество экземпляров[1]
	Основная литература[2]	
1	Сердюк Ю. П., Петров А. В. Параллельное программирование для многоядерных процессоров М.: Интернет-Университет Информационных Технологий, 2009	ЭБС «Университетская библиотека онлайн»
2	Афанасьев К. Е., Стуколов С. В., Малышенко В. В., Карабцев С. Н., Андреев Н. Е. Основы высокопроизводительных вычислений: учебное пособие. Т. 2. Технологии параллельного программирования Кемерово: Кемеровский государственный университет, 2012	ЭБС Университетская библиотека ONLINE
3	Федотов И. Е. Модели параллельного программирования М.: СОЛОН-ПРЕСС, 2012	ЭБС Университетская библиотека ONLINE
4	Богачёв К. Ю. Основы параллельного программирования: учебное пособие М.: БИНОМ. Лаборатория знаний, 2013	ЭБС Университетская библиотека ONLINE
5	Дэвис А. Асинхронное программирование в C# 5.0 М.: ДМК Пресс, 2013	ЭБС Университетская библиотека ONLINE
	Дополнительная литература[3]	
1	Уильямс Э. Параллельное программирование на C++ в действии. Практика разработки многопоточных программ М.: ДМК Пресс, 2012	ЭБС Университетская библиотека ONLINE
2	Немнюгин С. А. Модели и средства программирования для многопроцессорных вычислительных систем М.: Интернет-Университет Информационных Технологий, 2010	ЭБС Университетская библиотека ONLINE
3	Гергель В. П. Теория и практика параллельных вычислений М.: Интернет-Университет Информационных Технологий, 2007	ЭБС Университетская библиотека ONLINE
4	Сердюк Ю. П. Кластерные вычисления М.: Интернет-Университет Информационных Технологий, 2008	ЭБС Университетская библиотека

[1] Указывается в соответствии с фактическим наличием в библиотеке РГЭУ (РИНХ). Для определения количества экземпляров следует воспользоваться программой «Книгообеспеченность», установленной на кафедре.

[2] Указывается не более пяти источников. Год издания должен соответствовать требованиям образовательного стандарта.

[3] Перечень дополнительной литературы не ограничен по числу источников и году издания.

		ONLINE
Методические разработки[4]		

6.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

№	Выходные данные
1	Параллельное программирование с помощью языка C# http://www.intuit.ru/studies/courses/5938/1074/info
2	Введение в методы параллельного программирования http://www.intuit.ru/studies/courses/1021/284/info
3	Параллельное программирование http://www.intuit.ru/studies/courses/1110/153/info
4	Основы параллельных вычислений http://www.intuit.ru/studies/courses/1091/293/info

6.3. Перечень программного обеспечения

№	Наименование программного обеспечения[5]
1	Microsoft Office 2007
2	Visual Studio Express Edition

6.4. Перечень информационно-справочных систем

№	Наименование информационно-справочных систем
1	http://www.iqlib.ru/ - Электронно-библиотечная система
2	eLibrary.ru - научная электронная библиотека
3	www.biblioclub.ru - Университетская библиотека онлайн

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Помещения для проведения всех видов работ, предусмотренных учебным планом, укомплектованы необходимой специализированной учебной мебелью и техническими средствами обучения. Для проведения лекционных занятий используется демонстрационное оборудование. Лабораторные занятия проводятся в компьютерных классах, рабочие места в которых оборудованы необходимыми лицензионными программными средствами и выходом в Интернет.

8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Методические указания по освоению дисциплины представлены в Приложении 2 к рабочей программе дисциплины.

[4] Указываются методические разработки кафедры. Например, курсы лекций, указания по выполнению контрольной работы, указания по изучению дисциплины и т.д.
[5] Указываются только лицензионные программные средства

Приложение 1

Фонд оценочных средств

Защита лабораторных работ

Лабораторные работы №1-2. Знакомство с многопоточной обработкой

Вопросы

1. Почему эффект от распараллеливания наблюдается только при большем числе элементов?
2. Почему увеличение сложности обработки повышает эффективность многопоточной обработки?
3. Какое число потоков является оптимальным для конкретной вычислительной системы?
4. Почему неравномерность загрузки потоков приводит к снижению эффективности многопоточной обработки?
5. Какие другие варианты декомпозиции позволяют увеличить равномерность загрузки потоков?
6. В какой ситуации круговая декомпозиция не обеспечивает равномерную загрузку потоков?

Лабораторные работы №3-4. Поиск простых чисел

Вопросы и упражнения

1. Какими достоинствами и недостатками обладает каждый вариант распараллеливания?
2. Какие средства синхронизации можно использовать вместо конструкции lock? Какой вариант будет более эффективным?
3. Какой вариант ожидания завершения работ, запущенных пулом потоков, более эффективный и почему?
4. Реализуйте один или несколько вариантов распараллеливания с помощью объектов Task и с помощью метода Parallel.For. Выполните эффективность алгоритмов.
5. Реализуйте алгоритм поиска простых чисел как LINQ-запрос к массиву чисел.

Лабораторные работы №5-6. Синхронизация доступа к одноэлементному буферу

Вопросы и упражнения

1. Почему проблема гонки данных проявляется не при каждом прогоне?
2. Какие факторы увеличивают вероятность проявления проблемы гонки данных?
3. Возможно ли в данной задаче при отсутствии средств синхронизации возникновение исключения и аварийное завершение программы?
4. Можно ли в данной задаче использовать атомарные операторы для обеспечения согласованности доступа? Необходимы ли при этом дополнительные средства синхронизации?
5. Можно ли в данной задаче использовать потокобезопасные коллекции для обеспечения согласованного доступа?
6. Какие средства синхронизации обеспечивают наилучшее быстродействие в данной задаче? Объясните с чем это связано.

Лабораторные работы №7-9. Синхронизация приоритетного доступа к многоэлементному буферу

Вопросы и упражнения

1. Можно ли вместо объектов ManualResetEventSlim использовать другие типы сигнальных сообщений: AutoResetEvent или ManualResetEvent?
2. Какие особенности задачи не позволяют использовать объект ReaderWriterSlim?
3. Почему структура кольцевого буфера не требует синхронизации при работе одного читателя и одного писателя?
4. Почему в предложенной реализации не используются критические секции?
5. Реализуйте учет времени обращения рабочих потоков к буферу.

6. Реализуйте решение задачи с использованием конкурентных коллекций в качестве буфера.

Лабораторные работы №10-12. Клеточная модель "Игра Жизнь" Дж.Конвея

Вопросы и упражнения

1. Имеет ли смысл распараллеливание внутреннего цикла расчета? Почему?

```
for(int i=0; i < Height; i++)
{
 Parallel.For(0, Width, j =>
  {
   p = CalcPotential(table[i,j]);
     tableNew[i, j] = lifeRules(p, table[i, j]);
  });
}
```

2. Как вариант расчета – по строкам или по столбцам – более эффективен и с чем это связано?
3. Продумайте вариант блочной декомпозиции, где блок выступает матрицей размера $K \times K$. В чем достоинства и недостатки блочной декомпозиции для этой задачи? Какое значение параметра K следует выбирать?

Дополнительные задания для текущего контроля

Вопросы

В каких случаях не возникает проблема гонки данных

(Ответ считается верным, если отмечены все правильные варианты ответов.)

Вариант 1 ☐ один поток осуществляет запись в разделяемую переменную типа double, другой поток читает значения

Вариант 2 ☐ потоки осуществляют запись в разделяемую переменную типа int

Вариант 3 ☐ один поток добавляет элементы в коллекцию типа FIFO (очередь), другой поток извлекает элементы из коллекции

Вариант 4 ☐ потоки инкрементируют значения, которые хранятся в разделяемом массиве; каждый поток работает с элементом по уникальному индексу

Выберете правильные утверждения, соответствующие системам с общей памятью

(Ответ считается верным, если отмечены все правильные варианты ответов.)

Вариант 1 ☐ обеспечивается параллелизм только на уровне приложений как разделение процессорного времени

Вариант 2 ☐ для взаимодействия параллельных частей требуется применять средства синхронизации

Вариант 3 ☐ для взаимодействия параллельных потоков не требуется специальных интерфейсов передачи сообщений

Вариант 4 ☐ существует возможность параллелизма как на уровне потоков одного процесса, так и на уровне процессов

Выберете правильные утверждения, соответствующие системам с общей памятью

(Ответ считается верным, если отмечены все правильные варианты ответов.)

Вариант 1 при выполнении параллельной программы может возникать ситуация гонки данных

Вариант 2 применение средств синхронизации позволяет решить проблему гонки данных

Вариант 3 синхронизация необходима только при выполнении программы в режиме разделения времени

Вариант 4 для реализации параллельного выполнения программы обязательно применение средств синхронизации

Выберете правильные утверждения, соответствующие свойству масштабируемости параллельного приложения

(Отметьте один правильный вариант ответа.)

Вариант 1 приложение может обрабатывать разные объемы данных

Вариант 2 нет правильных вариантов

Вариант 3 приложение обеспечивает ускорение, равное предельному, согласно закону Амдала

Вариант 4 приложение максимально задействует все имеющиеся вычислительные ресурсы

Параллельный алгоритм состоит из четырех независимых подзадач, которые выполняются за одинаковое время. Ускорение алгоритма на двуядерной системе составляет 1.5. Чему равно ускорение алгоритма на четырех ядерной системе?

(Отметьте один правильный вариант ответа.)

Вариант 1 3

Вариант 2 2

Вариант 3 1.5

Вариант 4 6

Какое предельное ускорение может быть получено согласно закону Амдала для параллельной программы, содержащей 40% параллельного кода

(Отметьте один правильный вариант ответа.)

Вариант 1 10

Вариант 2 нет правильных ответов

Вариант 3 4

Вариант 4 5

Вопросы

1. Какие достоинства у непосредственной работы с потоками?
2. В чем заключаются основные проблемы организации многопоточной обработки?
3. В каких случаях работа с пулом потоков является более эффективной, чем непосредственная работа с потоками?

Упражнения

Реализуйте пользовательский пул потоков, который повторяет функциональность объекта ThreadPool. Добавьте новые методы пула, которые упрощают работу и делают общение с пулом более информативным. Например, методы ожидания запущенных задач; получение сводной информации о выполнении задач (число потоков, участвовавших в обработке; время обработки).

Вопросы

1. Можно ли организовать работу нескольких потоков без средств синхронизации?
2. Может ли многопоточное приложение, использующее только конструкции lock, войти в состояние взаимоблокировки?
3. Какие средства синхронизации позволяют реализовать функциональность критической секции? В каких случаях следует отдавать предпочтение тому или иному объекту?

Упражнения

1. Исследуйте эффективность легковесных средств синхронизации по сравнению с аналогичными объектами ядра операционной системы: SemaphoreSlim – Semaphore, ManualResetEvent – ManualResetEventSlim.
2. Для анализа можно использовать задачу обращения к разделяемому счетчику:

```
voidThreadFunc() {
// Вход в критическую секцию с помощью
// того или иного средства синхронизации
totalCount++;
// Выход из критической секции
}
```

3. Исследуйте эффективность потокобезопасных коллекций по сравнению с синхронизированным доступом к обычным коллекциям.
4. Исследуйте эффективность атомарных операторов по сравнению со средствами организации критической секции (lock, Monitor, Mutex).
5. Самостоятельно освойте работу с объектами, реализующими типовые схемы синхронизации, CountdownEvent и Barrier. Реализуйте функциональность этих объектов с помощью средств синхронизации, рассмотренных в лекции. Исследуйте эффективность и удобство работы объектов CountdownEvent и Barrier.

Вопросы

1. В чем основные достоинства работы с задачами по сравнению с непосредственной работой с потоками?
2. В каком случае задача завершается со статусом RanToCompletion при возникновении исключения?
3. Каким образом можно реализовать критическую секцию, используя задачи и не используя средств синхронизации, рассмотренных в предыдущей главе?
4. Для каких целей можно использовать вложенные недочерние задачи?

Упражнения

1. Исследуйте эффективность работы большего числа задач по сравнению с работой большого числа потоков.

Вопросы

1. Почему для досрочного выхода из параллельного цикла используются два метода Break и Stop?
2. В чем преимущество пакетной обработки?

Упражнения

1. Исследуйте эффективность цикла Parallel.For в задаче матричного умножения.

2. Исследуйте эффективность цикла Parallel.For с пакетной обработкой и без пакетной обработки.

Вопросы

1. Почему LINQ-запросы не распараллеливаются автоматически?
2. С какой целью используется оператор возвращения к последовательному выполнению запроса AsSequential?
3. Почему при выполнении параллельного запроса порядок элементов может сохраниться?

Упражнения

1. Составьте запросы, которые демонстрируют неэффективность статической декомпозиции.
2. Исследуйте эффективность выполнения PLINQ-запросов и шаблона Parallel.For.
3. Составьте запрос, с помощью которого можно убедиться в параллельности или последовательности его выполнения.
4. Исследуйте эффективность выполнения запроса с разными режимами буферизации.

Вопросы

1. В каком потоке будут выполняться вложенные задачи, если родительская задача запущена с опцией LongRunning?
2. Возможна ли ситуация блокировки потока при ожидании завершения задачи, находящейся в глобальной очереди?
3. В каких случаях возможно параллельное выполнение вложенных задач?
4. В каких случаях нарушается порядок выполнения вложенных задач?

Упражнения

1. Напишите код, который позволяет оценить достоинства и/или недостатки организации локальной очереди потока по принципу LIFO.
2. Напишите код с вычислительно-емкими задачами и оцените эффективность выполнения задач в пуле потоков и в сторонних потоках.

Приложение 2

Методические указания по освоению дисциплины

Основное внимание при изучении дисциплины необходимо сконцентрировать на прикладном аспекте ее использования. При реализации программы дисциплины используются различные образовательные технологии с учетом внедрения инновационных приемов и способов обучения при одновременном использовании традиционных методик.

Лекционный курс содержит теоретический материал, снабженный примерами. В процессе лекционного занятия студенты слушают преподавателя, задают вопросы, часть информации конспектируют. Лекционные занятия дополняются демонстрацией слайдов, концентрирующих внимание слушателей на ключевых моментах лекционного материала. Проведение лекционных занятий осуществляется в диалоговой форме с постановкой проблемных вопросов, допускающих возникновение дискуссий, что предполагает активное включение студентов в образовательный процесс. Лекция имеет цель – систематизация основы научных знаний по дисциплине, сконцентрировать внимание студентов на наиболее сложных и узловых проблемах.

При проведении практических занятий создаются условия для максимально самостоятельного выполнения заданий. Практические занятия предлагается выполнять на основе примеров с визуализацией всех заданий, для нестандартных задач предлагается дискуссия с «мозговым штурмом», применяется программированное обучение.

Текущая самостоятельная работа направлена на углубление и закрепление знаний студентов, развитие практических умений и представляет собой работу с лекционным материалом, поиск и обзор литературы и электронных источников информации по темам курса; выполнение индивидуальных заданий; опережающую самостоятельную работу; изучение тем, вынесенных на самостоятельную проработку; подготовку к семинарским занятиям; подготовку к контрольным работам.

Примеры интерактивных форм и методов проведения занятий:

Метод проектов

Метод проектов – это совокупность учебно-познавательных приемов, которые позволяют решить ту или иную проблему в результате самостоятельных действий студентов с обязательной презентацией этих результатов.

В основе метода проектов лежит развитие познавательных навыков студентов, умений самостоятельно конструировать свои знания, ориентироваться в информационном пространстве, развитие критического и творческого мышления. Для него характерны следующие приемы: определение источников информации; способов ее сбора и анализа, а также установление способа представления результатов (формы отчета). Устанавливаются процедура и критерии оценки результата и процесса разработки проекта, обязательное распределение заданий и обязанностей между членами команды.

Метод проектов всегда ориентирован на самостоятельную деятельность обучающихся - индивидуальную, парную, групповую, которую они выполняют в течение определенного отрезка времени. С другой стороны метод проектов – это совместная деятельность преподавателя и студента, направленная на поиск решения возникшей проблемы. Метод проектов всегда предполагает решение какой-то проблемы.

Метод проектов позволяет удачно сочетать черты исследовательского, творческого, информационного проекта и одновременно ориентирован на междисциплинарные связи.

Проекты подразделяются на: научные, обучающие, сервисные, социальные, творческие, рекламно-презентационные.

Как правило, в университетах наиболее привычными являются научные проекты, одним из вариантов которых можно считать курсовые и дипломные работы, а также некоторые виды активности в рамках учебных и производственных практик.

Разработка курсового или дипломного проекта (работы) решает проблему взаимосвязи теории и практики, придает профессиональную направленность обучению и повышает его качество. Кроме того, защита проекта с обязательным предложением конкретных конструктивных рекомендаций учит студентов технологически грамотно, логично излагать свои мысли, убеждать в необходимости и правильности принятого решения.

Изучение теоретического материала проводится в форме самостоятельной работы на опережающей основе. Это означает, что студенты изучают тему программы, которая ранее могла быть не представлена на занятии, при помощи учебников, справочников, литературы, источников, образовательных Интернет-ресурсов. (См. ПРИЛОЖЕНИЕ).

Методы группового решения творческих задач:

«Метод Дельфи».

«Метод Дельфи» - это не что иное, как разновидность мозгового штурма, метод быстрого поиска решений.

Целью этой технологии является получение согласованной информации высокой степени достоверности в процессе анонимного обмена мнениями между участниками группы экспертов для принятия решения.

«Метод Дельфи» позволяет учесть независимое мнение всех участников группы экспертов по обсуждаемому вопросу путем последовательного объединения идей, выводов и предложений и прийти к согласию. Метод основан на многократных анонимных групповых интервью.

На занятии преподаватель делит группу студентов на две подгруппы: так называемые «рабочую группу» для сбора и обобщения мнений экспертов, и экспертную группу, которая состоит «специалистов», владеющих вопросами по обсуждаемой теме. Далее следует подготовить анкету, указав в ней поставленную проблему, уточняющие вопросы. Формулировки должны быть четкими и однозначно трактуемыми, предполагать однозначные ответы. Затем провести опрос экспертов в соответствии с методикой, предполагающей при необходимости повторение процедуры. Полученные ответы служат основой для формулирования вопросов для следующего этапа. И, наконец, обобщить экспертные заключения и выдать рекомендации по поставленной проблеме.

Следует отметить, что достоинства этого метода состоят в том, что метод Дельфи способствует выработке независимости мышления членов группы; обеспечивает спокойное и объективное изучение проблем, которые требуют оценки. А недостатками метода являются чрезмерная субъективность оценок; подготовка к проведению занятий такого типа требует достаточно много времени и организационных усилий.

Контекстное обучение:

Сущность контекстного обучения определяется как организация такой деятельности, которая с необходимостью требует приобретения новых знаний и их последующего применения, объясняет и оправдывает усилия, затраченные на их усвоение. На основе эксперимента было установлено, что на развитие ключевых компетенций большое влияние оказывает контекстное обучение. Одним из средств контекстного обучения являются контекстные задачи.

Контекстная задача – это задача мотивационного характера, в условии которой описана конкретная жизненная ситуация, коррелирующая с имеющимся социокультурным опытом учащихся; требованием задачи является анализ, осмысление и объяснение этой ситуации или выбор способа действия в ней, а результатом решения задачи является встреча с учебной проблемой и осознание ее личностной значимости.

Чтение текста подобных задач понимается как деятельность имеющая значение не только для развития мыслительных операций учащегося, но и для приобретения им духовного и жизненного опыта.

При разработке контекстных задач необходимо учитывать несколько принципов.

1. Принцип доступности подразумевает, что для выполнения таких заданий не требуется фактических знаний за пределами программного материала. Это важно, потому что во время контроля знаний на уроке, на экзамене использование справочной литературы не допускается. Необходимо оценить уровень развития ключевых компетенций школьников на основе той знаниевой базы, которая заложена в образовательный стандарт. Свою эрудицию учащиеся, которые проявляют особые способности и устойчивый интерес к изучению химии, могут продемонстрировать при подготовке и проведении олимпиад, на внеклассных занятиях.
2. Принцип актуальности практически реализуется при использовании материалов СМИ. Обращение к новостям науки, анализу текущих событий в мире с точки зрения изучаемого предмета позволяет избежать искусственности заданий, что существенно влияет на рост мотивации при работе над ними.
3. Принцип учета возрастных особенностей учащихся. Предполагаемое задание должно быть интересно школьнику, а не вызывать скуку, описывая надуманную ситуацию.

Как правило, в контекстных заданиях содержатся вопросы и проблемы, с которыми ученик сталкивается в своей обыденно-практической жизни, литературных источниках, либо они соответствуют его профессиональным интересам и найдут применение в дальнейшем обучении.

Содержание традиционных и контекстных заданий направлено на контроль усвоения одних и тех же элементов знаний. Однако контекст заданий второго типа способен мотивировать ученика на поиск ответа на поставленную задачу, вызывать интерес с практической точки зрения и создать условия для применения знаний в ситуациях, способных возникать в реальной жизни. Контекстные задания могут предполагать самостоятельный поиск недостающей для решения информации, ее обобщение и анализ, то это позволяет оценивать показатели сформированности качества знаний учащихся.

Контекстные задачи, также как и контрольно-диагностические задания, компетентностно-ориентированные тесты используют как методы контроля достижений учащихся наряду с традицонными (контрольными работами, устными опросами)

Методы контроля достижений учащихся

/ /

Традиционные (контрольные работы, устные опросы)

Инновационные
Контрольно-диагностические задания
Компетентностно-ориентированные тесты
Контекстные задачи

Измерить уровень сформированности компетенций при помощи инновационного инструментария можно при условии выделения компонентов, контролируемых с помощью каждого задания: знаниевого, деятельностного (освоенные способы действия) и мотивационного.

Компьютерные симуляции:

Компьютерные симуляции - это максимально приближенная к реальности имитация процессов управления и принятия решений. Участники управляют компанией, выбирают стратегические цели для ее развития, принимают пошаговые тактические решения,

выстраивают взаимоотношение с клиентами, персоналом и акционерами, т.е. решают все те задачи по управлению, с которыми сталкиваются топ-менеджеры любой крупной корпорации.

Изначально этот метод активно применялся на Западе, однако наравне с case studies он пришел и в российские бизнес-школы. Данный подход близок к деловой игре и представляет собой виртуальную симуляцию различных явлений в бизнесе.

Лист переутверждения рабочей программы учебной дисциплины

Рабочая программа пересмотрена, обсуждена и одобрена для исполнения в 2017 - 2018 учебном году на заседании кафедры информатики, протокол № 10 от «04» мая 2017 г.

Зав. кафедрой информатики Ромм Я.Е.

Рабочая программа пересмотрена, обсуждена и одобрена для исполнения в 2017 - 2018 учебном году на заседании совета факультета физики, математики, информатики, протокол № 22 от «11» мая 2017 г.

Декан факультета физики, математики, информатики Донских С.А.