

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ
Проректор по учебной работе и
образовательным инновациям
О.И. Чуприс
« 17 » 12 2018 г.
Регистрационный № УД- 5593/уч.



Параллельные и распределенные вычисления

Учебная программа учреждения высшего образования
по учебной дисциплине для специальности

1-31 03 04 Информатика

2018 г.

Учебная программа составлена на основе ОСВО 1-31 03 04-2013 и учебных планов УВО № G31-169/уч. 2013 г. от 30.05.2013, № G31и-192/уч от 30.05.2013

СОСТАВИТЕЛЬ

Н. А. Лиходед, профессор кафедры вычислительной математики Белорусского государственного университета, доктор физико-математических наук, профессор.

РЕЦЕНЗЕНТЫ:

В. Б. Малютин, ведущий научный сотрудник отдела нелинейного и стохастического анализа Института математики НАН Беларуси, доктор физико-математических наук;

Н. И. Юрчук, профессор кафедры математической кибернетики механико-математического факультета Белорусского государственного университета, доктор физико-математических наук, профессор.

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой вычислительной математики (протокол № 14 от 19.04.2018);

Научно-методическим Советом БГУ (протокол № 5 от 04.05.2018)

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Учебная программа по учебной дисциплине «Параллельные и распределенные вычисления» разработана в соответствии с учебным планом и образовательным стандартом первой степени высшего образования по специальности 1-31 03 04 «Информатика».

Учебная дисциплина «Параллельные и распределенные вычисления» знакомит студентов со способами организации параллельных и распределенных вычислительных процессов, с основными подходами к решению важнейших задач отображения алгоритмов, задаваемых последовательными программами, на многоядерные персональные компьютеры и параллельные компьютеры с распределенной памятью. Рассматриваются, в частности, следующие задачи: получение информационной структуры алгоритма, получение и использование параллельных циклов, улучшение локальности вычислений, выбор зерна вычислений, организация параллельных и распределенных вычислительных процессов, построение параллельных алгоритмов, связанных с графами и сетями, аффинные преобразования гнезд циклов, автоматизация распараллеливания.

Учебная дисциплина «Параллельные и распределенные вычисления» относится к циклу специальных дисциплин, компонент УВО.

Основой для изучения учебной дисциплины являются следующие учебные дисциплины первой степени высшего образования: «Вычислительные методы алгебры», «Методы численного анализа», «Алгоритмы и структуры данных», «Архитектура компьютеров».

Цель преподавания учебной дисциплины «Параллельные и распределенные вычисления»: овладеть идеологией разработки параллельных алгоритмов на основе последовательных, получить навыки использования сведений по методам и алгоритмам параллельных и распределенных вычислений; формирование составной части знаний, получаемых будущими специалистами в процессе учебы и необходимых им в дальнейшем для успешной работы; формирование у студентов основ математического мышления, изучение различных подходов к построению вычислительных алгоритмов, ориентированных на параллельные компьютеры.

Основные задачи, решаемые при изучении учебной дисциплины «Построение и анализ параллельных алгоритмов»:

- введение студентов в проблематику организации параллельных и распределенных вычислений, введение в проблематику статического распараллеливания, основанного на знании информационной структуры программ;
- изучение основных понятий, математического аппарата и моделей параллельных вычислений;
- получение теоретических и практических основ выявления параллелизма, распараллеливания алгоритмов, преобразования последовательных программ в параллельные.

В результате изучения дисциплины обучаемый должен:

знать:

- проблематику статического распараллеливания;
- основные термины и понятия, математический аппарат и модели параллельных и распределенных вычислений;
- теоретические основы организации параллельных и распределенных вычислительных процессов, распараллеливания алгоритмов, преобразования последовательных программ в параллельные;

уметь:

- обнаруживать параллелизм;

- распределять операции и данные алгоритма между процессорами;
- устанавливать порядок выполнения операций и обмена данными.

владеть:

- основами разработки параллельных программ для многоядерных персональных компьютеров и компьютеров с распределенной памятью;
- инструментальными и программными средствами для организации параллельных и распределенных вычислительных процессов.

Освоение учебной дисциплины «Параллельные и распределенные вычисления» должно обеспечить формирование следующих академических, социально-личностных и профессиональных компетенций:

академические компетенции:

- АК-1. Уметь применять базовые научно-теоретические знания для решения теоретических и практических задач.
- АК-2. Владеть системным и сравнительным анализом.
- АК-3. Владеть исследовательскими навыками.
- АК-4. Уметь работать самостоятельно.

социально-личностные компетенции:

- СЛК-1. Обладать способностью к межличностным коммуникациям.

профессиональные компетенции:

- ПК-1. Работать с научно-технической, нормативно-справочной и специальной литературой.

В соответствии с типовым учебным планом специальности 1-31 03 04 Информатика учебная программа предусматривает для изучения дисциплины 158 учебных часов, в том числе 68 аудиторных часов: лекции – 34 часа, лабораторные занятия – 30 часов, управляемая самостоятельная работа – 4 часа.

Трудоемкость учебной дисциплины составляет 4 зачетные единицы.

Форма текущей аттестации – экзамен и зачет в 7 семестре.

Форма обучения – дневная.

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

Раздел 1. Введение в параллельные вычисления. Информационная структура алгоритма.

Тема 1.1. Основные понятия параллельных вычислений

Основная цель и задача параллельных вычислений. Терминология параллельных вычислений. Блочные матрицы. Блочные алгоритмы. Параллельные точечные и блочные алгоритмы для реализации на многоядерном процессоре.

Тема 1.2. Анализ зависимостей

Функции, задающие индексы массивов данных. Зависимости. Представление зависимостей. Функции зависимостей. Графы зависимостей. Типы зависимостей. Способы устранения ложных зависимостей. Параллельные и последовательные циклы.

Тема 1.3. Инструментальные средства для получения информационной структуры алгоритма

Инструментальные средства для получения информационной структуры алгоритма. Визуализация информационной структуры алгоритма, получение информации о зависимостях, типах зависимостей, получение функций зависимостей и графов зависимостей с помощью инструментальных средств.

Раздел 2. Организация параллельных и распределенных вычислений.

Тема 2.1. Тайлинг

Разбиение множества операций алгоритма на макрооперации-тайлы. Техника тайлинга. Допустимость тайлинга.

Тема 2.2. Организация параллельных вычислительных процессов. Параллельные последовательности зернистых вычислений

Задание зерна вычислений. Функции, задающие распределение зерен между процессорами. Условия параллельности последовательностей зернистых вычислений. Разработка и реализация на суперкомпьютере параллельных алгоритмов перемножения матриц, решения систем линейных алгебраических уравнений, решения дифференциальных уравнений в частных производных (распараллеливание, задание зерна вычислений, организация обмена данными).

Задачи (этапы) статического распараллеливания: получение информационной структуры алгоритма, выявление потенциального параллелизма алгоритма, выбор зерна вычислений, распределение входных данных, выделение массивов, согласование распределения операций и данных, организация обмена данными, улучшение локальности.

Тема 2.3. Параллельные алгоритмы, связанные с графами и сетями. Модель вычислений MapReduce

Параллельные алгоритмы обработки графов: поиск в ширину, нахождение кратчайшего пути, выделение компонент связности неориентированного графа, поиск множества путей между двумя вершинами, длина которых лежит в заданном диапазоне. Модель вычислений MapReduce, система Hadoop.

Раздел 3. Параллельная структура алгоритма. Инструментальные средства для

распараллеливания алгоритмов.

Тема 3.1. Распараллеливание, использующее уровни зависимостей

Распараллеливание, использующее уровни зависимостей. Алгоритм Аллена и Кеннеди распараллеливания гнезд тесно вложенных циклов.

Тема 3.2. Таймирующие функции

Таймирующие функции. Строгие таймирующие функции. Расщепляющие функции. Скошенный параллелизм. Векторные таймирующие функции.

Тема 3.3. Получение таймирующих функций для однородных гнезд циклов

Получение параметров таймирующих расщепляющих, строгих таймирующих функций посредством решения вспомогательных систем неравенств и уравнений. Алгоритм получения таймирующих функций.

Тема 3.4. Аффинные преобразования гнезд вложенных циклов

Преобразования тесно вложенных циклов, задаваемые векторными таймирующими функциями. Условия параллельности внутренних циклов после преобразования. Условия параллельности внешних циклов после преобразования.

Тема 3.5. Генерация кода. Инструментальные средства для получения параллельных алгоритмов.

Генерация кода: запись алгоритма после аффинного преобразования. Инструментальные средства для получения параллельных алгоритмов (в том числе и для генерации кода). Системы ParProc, LooPo, CLoog, PLUTO.

Раздел 4. Некоторые теоретические аспекты параллельных вычислений

Тема 4.1. Параллельная форма алгоритма. Концепция неограниченного параллелизма. Основные классы современных вычислительных систем

Параллельные множества операций, параллельная форма алгоритма, параллельные последовательности вычислений. Параллельные вычислительные свойства и параллельная структура алгоритма. Концепция неограниченного параллелизма. Принцип сдваивания. Процесс рекуррентного сдваивания. Основные классы современных параллельных компьютеров. Классификация Флинна параллельных компьютеров.

Тема 4.2. Характеристики параллельных вычислительных процессов и систем. Списки Top 500 и Graph 500

Обоснование определения понятий производительности, загруженности, ускорения вычислительной системы. Формулы для нахождения максимальных характеристик. Законы Амдала. Сетевой закон Амдала. Закон Густавсона-Барсиса. Сравнительный анализ законов Амдала и Густавсона-Барсиса. Список Top 500. Список Graph 500.

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ

Номер раздела, темы, занятия	Название раздела, темы, занятия; перечень изучаемых вопросов	Количество аудиторных часов		Количество часов УСП	Форма контроля знаний
		лекции	лабораторные занятия		
1	2	3	5	6	9
1.	Введение в параллельные вычисления. Информационная структура алгоритма (16 ч.)	6	9	1	Контрольная работа по теме 1.2, коллоквиум
1.1.	Основные понятия параллельных вычислений (1 ч.) Блочные матрицы. Блочные алгоритмы (7 ч.)	3	4		
1.1.1.	Цель и задача параллельных вычислений. Терминология параллельных вычислений.	1			
1.1.2.	Блочные матрицы. Операции над блочными матрицами. Параллельные циклы. Параллельные точечные и блочные алгоритмы для реализации на многоядерном процессоре.	2	4		Отчет по лаб. работе
1.2.	Анализ зависимостей (7 ч.)	3	3	1	
1.2.1.	Функции, задающие индексы массивов данных. Зависимости. Представление зависимостей. Функции зависимостей. Графы зависимостей. Типы зависимостей: истинные зависимости, антизависимости, зависимости по выходу, зависимости по входу.	2	2		Отчет по лаб. работе
1.2.2.	Способы устранения ложных зависимостей: переобозначение, увеличение размерности; расщепление операторов. Параллельные и последовательные циклы.	1			Устный опрос
1.2.3.	Получение информационной структуры алгоритма (получение векторов, функций и графов зависимостей).		1	1	Отчет по лаб. работе
1.3.	Инструментальные средства для получения информационной структуры алгоритма (2 ч.)		2		
1.3.1.	Система Loopo, система Depend grapher, Открытая распараллеливающая система. Визуализация информационной структуры алгоритма, получение информации о зависимостях, типах зависимостей, получение функций зависимостей и графов зависимостей с помощью инструментальных средств.		2		Отчет по лаб. работе
2.	Организация параллельных и распределенных вычислений. (26 ч.)	10	14	2	Контрольная работа по темам 2.1-2.3, коллоквиум
2.1.	Тайлинг (разбиение множества операций алгоритма на макрооперации) (4 ч.)	2	2		
2.1.1.	Разбиение множества операций алгоритма на тайлы. Примеры применения тайлинга. Техника тайлинга. Допустимость тайлинга.	2	2		Устный опрос
2.2.	Организация параллельных вычислительных процессов. Параллельные последовательности зернистых вычислений. (12 ч.)	4	6	2	
2.2.1.	Задание зерна вычислений. Функции, задающие распределение зерен между процессорами. Условия	2			Устный опрос

	параллельности последовательностей зернистых вычислений. Формулировка теоремы о допустимости тайлинга, применяемого для получения параллельных зернистых последовательностей вычислений.				
2.2.2.	Задачи (этапы) статического распараллеливания: получение информ. структуры алгоритма, выявление потенциального параллелизма алгоритма, выбор зерна вычислений, распределение входных данных, выделение массивов, согласование распределения операций и данных, организация обмена данными.	2			Устный опрос
2.2.3.	Разработка параллельных алгоритмов для выполнения матричных операций, решения систем линейных алгебраических уравнений, параллельных алгоритмов решения дифференциальных уравнений (реализация задач статического распараллеливания).		2	2	
2.2.4.	Реализация на суперкомпьютере параллельных алгоритмов.		4		Отчет по лаб. работе
2.3.	Параллельные алгоритмы, связанные с графами и сетями. Модель вычислений MapReduce (10 ч.)	4	6		
2.3.1.	Параллельные алгоритмы обработки графов: поиск в ширину, нахождение кратчайшего пути, выделение компонент связности неориентированного графа, поиск множества путей между двумя вершинами, длина которых лежит в заданном диапазоне.	4	4		
2.3.2.	Модель вычислений MapReduce, система Hadoop.		2		Отчет по лаб. работе
3.	Параллельная структура алгоритма. Инструментальные средства для распараллеливания алгоритмов (22 ч.)	14	7	1	Контрольная работа по темам 3.2-3.5, коллоквиум
3.1.	Распараллеливание, использующее уровни зависимостей (1 ч.)	1			
3.1.1.	Алгоритм Аллена и Кеннеди распараллеливания гнезд тесно вложенных циклов.	1			
3.2.	Таймирующие функции (1 ч.)	1			
3.2.1.	Таймирующие функции. Строгие таймирующие функции. Расщепляющие функции. Скошенный параллелизм. Векторные таймирующие функции. Многомерное таймирование.	1			
3.3.	Получение таймирующих функций для однородных гнезд циклов (2 ч.)	2			
3.3.1.	Получение параметров таймирующих, расщепляющих, строгих таймирующих функций посредством решения вспомогательных систем неравенств и уравнения. Алгоритм получения таймирующих функций.	2			Устный опрос
3.4.	Аффинные преобразования гнезд вложенных циклов (5 ч.)	3	2		
3.4.1.	Преобразования вложенных циклов, задаваемые векторными таймирующими функциями. Условия параллельности внутренних циклов после преобразования. Условия параллельности внешних циклов после преобразования.	3	2		Устный опрос
3.5.	Генерация кода. Инструментальные средства для получения параллельных алгоритмов (в том числе и для генерации кода) (13 ч.)	7	5	1	
3.5.1.	Генерация кода: запись алгоритма после аффинного преобразования.	2	2		
3.5.2.	Системы ParProc, LooPo, CLoog, PLUTO.	2	2		Отчет по лаб. работе
3.5.2.	Примеры получения параллельной структуры алгоритма и генерации кода.	3	1	1	
4.	Некоторые теоретические аспекты	4			Коллоквиум

	параллельных вычислений (4 ч.)				
4.1.	Параллельная форма алгоритма. Концепция неограниченного параллелизма. Основные классы современных вычислительных систем. (2 ч.)	2			
4.1.1.	Параллельные множества операций, параллельная форма алгоритма, параллельные последовательности вычислений. Параллельные вычислительные свойства и параллельная структура алгоритма. Суть и основная цель концепции неограниченного параллелизма. Принцип сдваивания. Процесс рекуррентного сдваивания.	1			
4.1.2.	Основные классы современных вычислительных систем. Классификация Флинна параллельных компьютеров.	1			
4.2.	Характеристики параллельных вычислительных процессов и систем. Списки Top 500 и Graph 500 (2 ч.)	2			
4.2.1.	Определения производительности, загруженности, ускорения вычислительной системы: Формулы максимальных производительности, загруженности, ускорения. Законы Амдала. Сетевой закон Амдала. Закон Густавсона-Барсиса. Сравнительный анализ законов Амдала и Густавсона-Барсиса. Список Top 500. Список Graph 500.	2			

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

Перечень основной литературы

1. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. – Санкт-Петербург. БХВ-Петербург. 2002. – 600 с.
2. Воеводин В. В. Вычислительная математика и структура алгоритмов. – Москва: Изд-во МГУ, 2006. – 112 с. <http://parallel.ru/info/parallel/voevodin/>
3. Лиходед Н.А. Методы распараллеливания гнезд циклов: Курс лекций. – Мн.: БГУ. 2008. – 100 с.
4. [Электрон. ресурс – Персональная страница Лиходеда Н.А. на www.bsu.by] Лекции.
[Электрон. ресурс – fpmi-stud/Subfaculty/Каф. Выч. Мат/Параллельные вычисления/Лекции]

Перечень дополнительной литературы

5. Гервич Л.Р., Штейнберг Б.Я., Юрушкин М.В. Разработка параллельных программ с оптимизацией использования структуры памяти. – Ростов-на-Дону. Изд-во Южного федерального университета, 2014. – 120 с.
6. Антонов А.С. Параллельное программирование с использованием технологии OpenMP. – Москва, МГУ. 2009. – 77 с. http://parallel.ru/tech/tech_dev/OpenMP/examples
7. [Электрон. ресурс – fpmi-stud/Subfaculty/Каф. Выч. Мат/Параллельные вычисления/Инструментарий] Система Depend grapher.
8. LooPo. www.fmi.uni-passau.de/loopo.
9. Открытая распараллеливающая система: www.ops.rsu.ru.
10. Kim D., Rajopadhye S. Parameterized tiling for imperfectly nested loops. // Technical Report CS-09-101, Colorado State University, Department of Computer Science, February 2009.
11. Tavarageri S., Hartono A., Baskaran M., Pouchet L.-N., Ramanujam J., Sadayappan, P. Parametric tiling of affine loop nests // Proc. 15th Workshop on Compilers for Parallel Computers. Vienna, Austria, July 2010.
12. Лиходед Н.А., Толстикова А.А. Параллельные последовательности зернистых вычислений // Доклады НАН Беларуси. 2010. Т. 54, № 4. С. 36–41.
13. Антонов А.С., Воеводин Вл.В. Эффективная адаптация последовательных программ для современных векторно-конвейерных и массивно-параллельных супер-ЭВМ. Программирование. 1996. № 4. С. 37–51.
14. Антонов А.С. Параллельное программирование с использованием технологии MPI. – Москва, МГУ. 2004. – 71 с. <http://parallel.ru/info/parallel/antonov/>
15. Воеводин Вл.В., Воеводин Вад.В. Спасительная локальность суперкомпьютеров // Открытые системы. 2013, № 9. С. 12–15.
16. Боресков А.В., Харламов А.А. Основы работы с технологией CUDA. – Москва. ДМК Пресс, 2010. – 232 с.
17. Baskaran M., Ramanujam J., Sadayappan, P. Automatic C-to- CUDA code generation for affine programs // Proceedings of the Compiler Construction, 19th International Conference. Part of the Joint European Conferences on Theory and Practice of Software. Paphos, Cyprus, March 20–28, 2010.
18. Гергель В.П. Теория и практика параллельных вычислений. – Москва: Интернет-Университет, БИНОМ. Лаборатория знаний, 2007.
19. Agarwal V., Petrini F., Pasetto D., Bader D. Scalable graph exploration on multicore processors // Proceedings of the ACM/IEEE Supercomputing 2010 Conference. New Orleans, LA, November 13–19, 2010.
20. Yoo A., Chow E., Henderson K., McLendon W., Hendrickson B., Catalyurek U.V: A scalable distributed parallel breadth-first search algorithm on BlueGene/L // Proceedings of the ACM/IEEE Supercomputing 2005 Conference. November 2005.
21. Buluc A., Madduri K.: Parallel breadth-first search on distributed memory systems // Proceedings of the ACM/IEEE Supercomputing 2011 Conference. November 2011.
22. Kang S., Bader D. Large scale complex network analysis using the hybrid combination of a MapReduce cluster and a highly multithreaded system // 4th Workshop on Multithreaded Architectures and Applications (MTAAP), Atlanta, GA, April 23, 2010.

23. Электрон. ресурс – \\fpmi-stud\Subfaculty\Каф. Выч. Мат\Параллельные вычисления\Лекции\ Система ParProc.
24. Библиотеки BLAS, LAPACK, BLACS, ScaLAPACK: <http://www.netlib.org>
25. PLUTO: pluto-compiler.sourceforge.net.
26. CLooG: The Chunky Loop Generator. <http://www.cloog.org>.

Примерный перечень заданий для управляемой самостоятельной работы студентов

Раздел 1. Информационная структура алгоритма.

1. Найти зависимости указанного алгоритма.
2. Найти функции зависимостей указанного алгоритма: вид функции, область определения.
3. Построить развернутый и редуцированный графы зависимостей указанного алгоритма.
4. Выяснить, образует ли однопараметрическую группу каждое из заданных преобразований.
5. Найти преобразования группы, задаваемой указанным оператором.

Раздел 2. Организация параллельных вычислений.

1. Указать распределение входных и выходных данных одного из вариантов параллельного алгоритма прямого хода метода Гаусса.
2. Произвести выделение и приватизацию массивов одного из вариантов параллельного алгоритма перемножения матриц.
3. Записать коммуникационные операции одного из вариантов параллельного алгоритма перемножения матриц.

Раздел 3. Параллельная структура алгоритма. Генерация кода.

1. Обосновать параллельность внутреннего цикла после применения к алгоритму заданного аффинного преобразования.
2. Обосновать параллельность внешнего цикла после применения к алгоритму заданного аффинного преобразования.
3. Сгенерировать псевдокод, получаемый после заданного аффинного преобразования.

Перечень рекомендуемых средств диагностики

Для текущего контроля качества усвоения знаний студентами используется следующий диагностический инструментарий:

- лабораторные работы;
- письменные контрольные работы;
- коллоквиумы;
- устные экспресс-опросы.

Лабораторные работы, как правило, представляют собой задания, включающие аналитическое исследование предложенной задачи, а также применения средств автоматизации получения программной реализации указанного параллельного алгоритма, проведение вычислительного эксперимента и комментарии по его итогам. Рекомендуемая форма отчетности по лабораторной работе – отчет. Лабораторная работа оценивается по стандартной 10-балльной шкале. Оценка за лабораторную работу может быть снижена в случае несвоевременного выполнения.

Письменные контрольные работы проводятся для контроля знаний по одному или двум разделам курса. Они включают два задания, разбитые на 5-6 подзаданий, и оцениваются по 10-балльной шкале. В случае неудовлетворительной оценки контрольная работа может быть переписана.

Коллоквиум представляет собой персональную устную беседу преподавателя со студентом с целью определения уровня знаний по пройденным темам. Для более точной оценки

коллоквиум может включать дополнительный письменный этап. По результатам коллоквиума выставляется оценка по 10-балльной шкале.

Устный экспресс-опрос студентов проводится в свободной форме в течение лабораторных занятий. Его результаты учитываются преподавателем при выставлении рейтинговой оценки в конце семестра.

Методика формирования итоговой оценки

Итоговая оценка формируется на основе:

1. Правил проведения аттестации студентов (Постановление Министерства образования Республики Беларусь №53 от 29 мая 2012 г.);
2. Положения о рейтинговой системе оценки знаний по дисциплине в БГУ (Приказ ректора БГУ от 18.08.2015)
3. Критериев оценки знаний студентов (письмо Министерства образования от 22.12.2003 г.)

ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ

Название учебной дисциплины, с которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы учреждения высшего образования по учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу
Современные методы разработки эффективных алгоритмов обработки больших объемов информации (II степень)	Кафедра дискретной математики и алгоритмики	Нет	Изменений не требуется, протокол № 14 от 19.04.2018
Технологии и компьютерные системы обработки больших объемов информации (II степень)	Кафедра дискретной математики и алгоритмики	Нет	Изменений не требуется, протокол № 14 от 19.04.2018

**ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К УЧЕБНОЙ ПРОГРАММЕ ПО ИЗУЧАЕМОЙ
УЧЕБНОЙ ДИСЦИПЛИНЕ**
на ____/____ учебный год

№ п/п	Дополнения и изменения	Основание

Учебная программа пересмотрена и одобрена на заседании кафедры
_____ (протокол № ____ от _____ 20__ г.)

Заведующий кафедрой

УТВЕРЖДАЮ
Декан факультета
