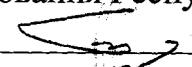


МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
Учебно-методическое объединение по естественнонаучному образованию

УТВЕРЖДАЮ

Первый заместитель Министра
образования Республики Беларусь


В.А.Богуш

« 20 » 08 2015г.

Регистрационный № ТД-С. 512/тип.

Теория алгоритмов

Типовая учебная программа по учебной дисциплине

для специальности

1-31 03 04 Информатика

СОГЛАСОВАНО

Председатель Учебно-методического объединения по естественнонаучному образованию

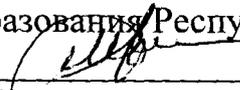

Голстик

« 20 » 08 2015 г.



СОГЛАСОВАНО

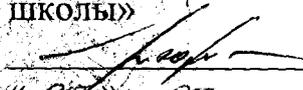
Начальник Управления высшего образования Министерства образования Республики Беларусь


С.И. Романюк

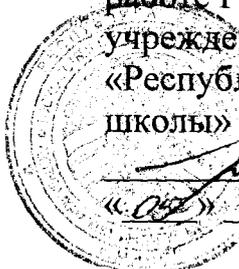
« 20 » 08 2015 г.

СОГЛАСОВАНО

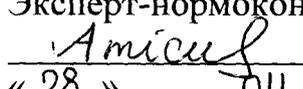
Проректор по научно-методической работе Государственного учреждения образования «Республиканский институт высшей школы»


И.В. Титович

« 05 » 05 2015 г.



Эксперт-нормоконтролер


А.А. Денисевич

« 28 » 04 2015 г.

Минск 2015

СОСТАВИТЕЛИ:

В.М. Котов, заведующий кафедрой дискретной математики и алгоритмики Белорусского государственного университета, доктор физико-математических наук, профессор;

Е.П. Соболевская, доцент кафедры дискретной математики и алгоритмики Белорусского государственного университета, кандидат физико-математических наук, доцент

РЕЦЕНЗЕНТЫ:

Кафедра программного обеспечения интеллектуальных и компьютерных систем Учреждения образования «Гродненский государственный университет имени Янки Купалы»;

В.М. Демиденко – профессор кафедры высшей математики Белорусского национального экономического университета, доктор физико-математических наук, профессор

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ В КАЧЕСТВЕ ТИПОВОЙ

Кафедрой дискретной математики и алгоритмики Белорусского государственного университета
(протокол № 13 от 16 апреля 2014г.);

Научно-методическим советом Белорусского государственного университета
(протокол № 5 от 15 мая 2014г.);

Научно-методическим советом по прикладной математике и информатике учебно-методического объединения по естественнонаучному образованию
(протокол № 7 от 22 апреля 2014г.).

Ответственный за редакцию: Е.П. Соболевская

Ответственный за выпуск: Е.П. Соболевская

Пояснительная записка

Типовая учебная программа по учебной дисциплине «Теория алгоритмов» разработана в соответствии с типовым учебным планом и образовательным стандартом первой ступени высшего образования по специальности 1-31 03 04 «Информатика»

Учебная дисциплина «Теория алгоритмов» знакомит студентов с фундаментальными понятиями, используемыми при разработке алгоритмов и оценке их качества. В типовую учебную программу включены разделы, позволяющие строить эффективные алгоритмы для разнообразных задач дискретной и комбинаторной оптимизации с использованием различных структур данных.

Основой для изучения учебной дисциплины «Теория алгоритмов» являются учебные дисциплины «Дискретная математика и математическая логика» государственного компонента и «Программирование» компонента учреждения высшего образования. Методы, излагаемые в учебной дисциплине «Теория алгоритмов», используются при изучении учебных дисциплин «Исследование операций», «Модели данных и системы управления базами данных» государственного компонента, а также при изучении ряда дисциплин специализации. Изучение учебной дисциплины позволяет дать студентам базу, необходимую для успешного усвоения материала перечисленных выше учебных дисциплин, а также получить знания, необходимые им в дальнейшем для успешной работы.

Цель преподавания учебной дисциплины «Теория алгоритмов» – формирование навыков для построения и анализа методов и алгоритмов при решении модельных задач дискретной оптимизации и их применение на практике.

При изложении материала учебной дисциплины целесообразно выделить этап построения математической модели, существенно влияющей на ее адекватность реальной проблеме, а также показать возможность использования аппарата теории алгоритмов для анализа и обоснования выбора наиболее эффективных методов и алгоритмов для решения прикладных задач.

Основные задачи, решаемые при изучении учебной дисциплины «Теория алгоритмов»:

- формирование у студентов таких фундаментальных понятий как информация, размерность задачи и трудоемкость алгоритмов;
- изучение подходов для определения трудоемкости алгоритмов посредством составления и решения рекуррентных уравнений;
- изучение современных структур данных и обоснование выбора соответствующей структуры в зависимости от набора базовых операций, используемых в алгоритме;
- изучение методов решения NP -трудных задач и оценка качества построенных решений.

В результате изучения учебной дисциплины студент должен **знать:**

- понятие размерности задачи и трудоемкости алгоритма;
- основные способы решения рекуррентных уравнений;

- основные подходы при разработке эффективных алгоритмов;
- способы организации структур данных и технологию их использования;
- виды поисковых деревьев;
- базовые алгоритмы на графах;
- основные методы отсечений при организации перебора вариантов;
- способы оценки приближенных алгоритмов;
- основные приемы, используемые для сжатия информации;
- основные модели параллельных вычислений;

уметь:

- сводить решение исходной задачи к решению подзадач и определять трудоемкость алгоритмов на основе рекуррентных соотношений;
- выбирать подходящие структуры данных при разработке эффективного алгоритма решения задачи;
- реализовывать поисковые деревья;
- строить графовые модели и применять базовые графовые алгоритмы;
- организовывать дерево перебора вариантов и осуществлять отсечения бесперспективных решений;
- оценивать качество приближенных алгоритмов;

владеть:

- основными подходами для разработки эффективных алгоритмов: метод «разделяй и властвуй» и динамическое программирование;
- навыками реализации и использования структур данных;
- способами организации перебора вариантов при решении *NP*-трудных задач;
- приемами построения и оценки качества приближенных алгоритмов.

Типовая учебная программа рассчитана на 208 часов, из них 136 аудиторных часов примерное распределение которых по видам занятий включает: 68 лекционных часов и 68 часов лабораторных занятий.

Рекомендуемые формы текущей аттестации – экзамен, зачет.

В соответствии с требованиями образовательного стандарта по специальности 1-31 03 04 «Информатика» специалист должен владеть следующими академическими компетенциями (АК) и профессиональными компетенциями (ПК):

АК-1. Уметь применять базовые научно-теоретические знания для решения теоретических и практических задач.

АК-6. Владеть междисциплинарным подходом при решении проблем.

ПК-14. Работать с научной, нормативно-справочной и специальной литературой.

Примерный тематический план

№ темы	Количество аудиторных часов	
	Лекции	Лабораторные занятия
Содержание		
Раздел I. Анализ алгоритмов		
1. Введение. Основные понятия и определения	2	
2. Рекуррентные уравнения и основные методы их решения	4	2
Раздел II. Разработка эффективных алгоритмов		
3. Динамическое программирование	1	4
4. Метод «разделяй и властвуй»	1	2
Раздел III. Структуры данных		
5. Простейшие структуры данных	2	2
6. Специализированные структуры данных	6	6
7. Система непересекающихся множеств	2	2
8. Хеш-таблицы	2	2
9. Поисковые деревья	4	4
Раздел IV. Графовые алгоритмы		
10. Способы обхода вершин графа	4	4
11. Кратчайший маршрут	2	2
12. Минимальное остовное дерево	2	2
13. Поток в сети	2	2
Раздел V. Организация перебора вариантов		
14. Построение дерева вариантов	2	8
15. Методы отсечений	4	8
Раздел VI. Приближенные алгоритмы		
16. Погрешность алгоритмов. Нижние оценки оптимального решения. Матроидные структуры. Разрешимые случаи.	8	8
17. Алгоритмы с гарантированной оценкой	4	6
Раздел VII. Параллельные алгоритмы		
18. Модели параллельных вычислений	4	1
19. Методы распараллеливания	8	1
Раздел VIII. Теория кодирования		
20. Алгоритмы на основе вероятности	2	1
21. Алгоритмы с учетом повторения	2	1
Всего часов	136	

Содержание учебного материала

Раздел I. Анализ алгоритмов

1. Введение. Основные понятия и определения

Предмет теории алгоритмов. Историческое развитие теории алгоритмов и ее место среди других математических наук и в естествознании.

Формальное описание задачи. Размерность задачи. Трудоемкость алгоритма. Асимптотики O , Ω , Θ . Полиномиальные, псевдополиномиальные и экспоненциальные алгоритмы. Примеры алгоритмов решения задач и оценка их трудоемкости.

2. Рекуррентные уравнения и основные методы их решения

Понятие рекуррентного уравнения. Полное рекуррентное уравнение. Основные методы решения рекуррентных уравнений: метод итераций, метод рекурсивных деревьев. Оценка решения рекуррентного уравнения: метод подстановок. Теорема о решении рекуррентного уравнения вида $T(n) = a \cdot T\left(\frac{n}{c}\right) + b \cdot n$.

Оценка трудоемкости базовых алгоритмов внутренней сортировки и поиска, используя рекуррентные уравнения.

Выбор способа программной реализации рекуррентного соотношения.

Раздел II. Разработка эффективных алгоритмов

3. Динамическое программирование

Основные подходы к разработке эффективных алгоритмов: динамическое программирование. Примеры решения задач.

4. Метод «разделяй и властвуй»

Основные подходы к разработке эффективных алгоритмов: метод «разделяй и властвуй». Примеры решения задач.

Раздел III. Структуры данных

5. Простейшие структуры данных

Способы организации базовых структур данных: массив, простой список, мультисписок, стек, очередь. Реализация базовых операций и их трудоемкость.

Технология использования простейших структур данных на примере алгоритма сжатия информации Хафмена.

6. Специализированные структуры данных

Приоритетная очередь. Бинарная куча. Биномиальная куча. Куча Фибоначчи. Реализация базовых операций и их трудоемкость. Амортизированная (усредненная) оценка трудоемкости операции. Примеры решения задач.

7. Система непересекающихся множеств

Система непересекающихся множеств. Различные способы представления системы непересекающихся множеств в памяти компьютера. Реализация базовых операций и их трудоемкость. Примеры решения задач.

8. Хеш-таблицы

Прямая адресация. Хеш-таблицы и хеш-функции. Коллизии. Методы разрешения коллизий: метод цепочек, открытая адресация. Универсальное семейство хеш-функций. Совершенное хеширование.

9. Поисковые деревья

Методы хранения деревьев в памяти компьютера. Бинарные поисковые деревья. Сбалансированные поисковые деревья: AVL-деревья, 2–3-деревья. Поддержка инвариантов сбалансированности. Реализация базовых операций и их трудоемкость.

Раздел IV. Графовые алгоритмы

10. Способы обхода вершин графа

Графовые модели. Методы хранения графов в памяти компьютера.

Алгоритм поиска в глубину в графе и его трудоемкость. Алгоритм поиска в ширину в графе и его трудоемкость. Связность, двудольность графа. Выделение сильно связанных компонент ориентированного графа. Маршруты, обладающие заданными свойствами. Топологическая сортировка. Эйлеров цикл.

11. Кратчайший маршрут

Алгоритмы построения кратчайших маршрутов в графе и их трудоемкость. Различные подходы к программной реализации алгоритма Дейкстры и их трудоемкость.

12. Минимальное остовное дерево

Минимальное остовное дерево графа. Алгоритм Прима. Алгоритм Краскала. Трудоемкость алгоритмов построения минимального остовного дерева.

13. Поток в сети

Максимальный поток в сети и его приложения.

Раздел V. Организация перебора вариантов

14. Построение дерева вариантов

Построение дерева вариантов. Способы обхода: односторонний и фронтальный.

15. Методы отсечений

Методы уменьшения перебора: отсечения по повторению, по недопустимости. Понятие рекорда и нижней оценки. Отсечения по рекорду.

Раздел VI. Приближенные алгоритмы

16. Погрешность алгоритмов. Нижние оценки оптимального решения. Матроидные структуры. Разрешимые случаи.

Построение приближенных решений. Погрешность алгоритмов. Анализ наихудшего случая. Градиентные алгоритмы. Матроидные структуры. Разрешимые случаи для задачи гамильтонова цикла. Теорема Дирака, Хватала.

17. Алгоритмы с гарантированной оценкой

Алгоритмы с гарантированной оценкой. ε -приближенные и быстрые ε -приближенные алгоритмы. Алгоритмы с неполной информацией. On-line и semi on-line версии задач (задачи с неполной информацией о входных данных), алгоритмы их решения и способы оценки качества решений.

Раздел VII. Параллельные алгоритмы

18. Модели параллельных вычислений

Сортирующие сети. Компараторы. Полуочиститель. Сети слияния. Модели параллельных вычислений. Понятие PRAM-машины. Различные типы машин: EREW PRAM, CREW PRAM, CRCW PRAM (Weak, Common, Arbitrary, Priority, Strong). Связь между ними.

19. Методы распараллеливания

Общие методы распараллеливания: метод сдваивания, матричная техника, сепараторы. Рандомизированные алгоритмы.

Раздел VIII. Теория кодирования

20. Алгоритмы на основе вероятности

Алгоритмы на основе вероятности. Алгоритм Хаффмена, Шенона. Арифметическое кодирование.

21. Алгоритмы с учетом повторения

Алгоритмы типа Лемпеля-Зива. Алгоритм Левенштейна.

Информационно-методическая часть

Литература

Основная

1. Ахо, А. В. Структуры данных и алгоритмы/ А. В. Ахо, Д. Э. Хопкрофт, Д. Д. Ульман. : Учеб. пособие/ пер. с англ. М. : Вильямс, 2000. 384 с.
2. Кормен, Т. Алгоритмы : построение и анализ/ Т. Кормен, Ч. Лейзерсон, Р. Ривест, К. Штайн. М. : Вильямс, 2005. 1296 с.
3. Ковалев, М. Я. Теория алгоритмов. Часть 2. Приближенные алгоритмы./ Ковалев М.Я., Котов В.М. Лепин В.В. – Мн.: БГУ, 2003. – 147 с.
4. Котов, В. М. Алгоритмы и структуры данных / В. М. Котов, Е. П. Соболевская, А. А. Толстикова – Минск : БГУ, 2011 – 267 с. – (Классическое университетское издание).

Дополнительная

5. Волчкова, Г. П. Сборник задач по теории алгоритмов для студентов физико-математических спец. БГУ/ Г. П. Волчкова, В. М. Котов, Е. П. Соболевская. Минск : БГУ, 2005. 59 с.
6. Волчкова, Г. П. Сборник задач по теории алгоритмов. Организация перебора вариантов и приближенные алгоритмы: для студентов спец. 1-31 03 04

- «информатика» / Г. П. Волчкова, В. М. Котов, Е. П. Соболевская. Минск : БГУ, 2008. 59 с.
7. *Вирт Н.* Алгоритмы и структуры данных. – СПб.: Невский Диалект, 2001. – 352 с.
 8. *Емеличев, В. А.* Лекции по теории графов/ В. А. Емеличев, О. И. Мельников, В. И. Сарванов, Р. И. Тышкевич. – М.: Наука, 1990. – 383 с.
 9. *Липский В.* Комбинаторика для программистов. – М.: Мир, 1988. – 214 с.
 10. *Пападдимитриу, Х.* Комбинаторная оптимизация: Алгоритмы и сложность/ Х. Пападдимитриу, К. Стайглиц. – М.: Мир, 1971. – 512 с.
 11. *Рейнгольд, Э.* Комбинаторные алгоритмы теория и практика/ Э. Рейнгольд, Ю. Нивергельт, Н. Део. – М.: Мир, 1980. – 476 с.
 12. Теория алгоритмов: учеб. пособие / П. А. Иржавский [и др.]. – Минск : БГУ, 2013. – 159 с.
 13. Mark Allen Weiss. Data structures and algorithm analysis. – Benjamin/Cummings Publishing Company, 1992. – 455 p.
 14. Shaffer C. A Practical Introduction to Data Structure and Algorithm Analysis. – London: Prentice Hall International, 1997. – 494 p.

Диагностика компетенций студента

На лекционных занятиях по учебной дисциплине «Теория алгоритмов» возможно использование элементов проблемного обучения: проблемное изложение некоторых аспектов, использование частично-поискового метода.

На лабораторных занятиях по учебной дисциплине «Теория алгоритмов» рекомендуется использовать индивидуальный, творческий подход. Студенты получают от преподавателя индивидуальные задания, в рамках самостоятельной работы разрабатывает свои алгоритмы их решения с последующими реализациями на некотором языке программирования и проверкой их работоспособности на компьютере, а также доказывают эффективность разработанных алгоритмов с точки зрения выбранных структур данных, трудоемкости и объема используемой памяти. Для организации самостоятельной работы студентов целесообразно использовать в учебном процессе разработанные компьютерные тесты.

Условия для самостоятельной работы студентов, в частности, для развития навыков самоконтроля, способствующих интенсификации учебного процесса, обеспечиваются:

- наличием и использованием в учебном процессе открытых систем автоматического тестирования, которые доступны пользователям через Интернет в любое удобное для них время (локальный доступ, удаленный доступ);
- в рамках самостоятельной работы обеспечение проверки на использование несанкционированных материалов;
- наличием и полным доступом обучающегося к библиотечным фондам, электронным средствам обучения, доступностью электронных (и бумажных)

вариантов лекций, учебно-методических пособий и сборников задач по основным разделам учебной дисциплины, примеров решения типовых задач.

Текущий контроль самостоятельной работы студентов по усвоению знаний по учебной дисциплине «Теория алгоритмов» рекомендуется осуществлять преподавателем на аудиторных занятиях в виде вопросов для самоконтроля, проведения коллоквиумов, письменных контрольных работ, проверки выполнения индивидуальных заданий.

В качестве рекомендуемых технических средств диагностики целесообразно использовать системы автоматического тестирования – инструмент с эффективной функциональностью контроля, тренинга и самостоятельной работы. Для контроля самостоятельности выполнения работ рекомендуется выдавать каждому студенту индивидуальные задания и использовать автоматические системы определения несанкционированных материалов.

Рекомендуемая форма текущей аттестации – экзамен, зачет.