

МИНОБРНАУКИ РОССИИ

Ярославский государственный университет им. П.Г. Демидова

Кафедра дискретного анализа

УТВЕРЖДАЮ

Декан факультета ИВТ

 Д.Ю. Чалый

«23» мая 2023 г.

Рабочая программа дисциплины

«Анализ алгоритмов и сложность вычислений»

Направление подготовки

02.04.02 Фундаментальная информатика и информационные технологии

Направленность (профиль)

«Искусственный интеллект и компьютерные науки»

Квалификация выпускника

Магистр

Форма обучения

очная

Программа рассмотрена
на заседании кафедры
от «11» апреля 2023 г.,
протокол № 4

Программа одобрена НМК
факультета ИВТ
протокол № 6 от
«28» апреля 2023 г.

Ярославль

1. Цели освоения дисциплины

Дисциплина «Анализ алгоритмов и сложность вычислений» относится к вариантивной части ОП магистратуры.

Для освоения данной дисциплины студенту необходимы знания дискретной математики, теории алгоритмов и анализа сложности алгоритмов.

Полученные в курсе «Анализ алгоритмов и сложность вычислений» знания необходимы для изучения обязательных дисциплин профессионального цикла магистратуры: «Теория информации».

2. Место дисциплины в структуре ОП магистратуры

Дисциплина «Анализ алгоритмов и сложность вычислений» относится к вариативной части Блока 1.

Основу курса составляет изучение современных концепций текстовой верстки на основе издательской системы LaTeX. Содержание курса концентрируется вокруг общей идеи отделения контента текстового документа от описания его представления. Предполагается, что знания и навыки, полученные студентами при изучении данного курса, будут использоваться ими при подготовке научных публикаций, а также при написании выпускной квалификационной работы. Кроме того, материал курса является ценным иллюстративным материалом для других курсов, в частности, для курсов «Алгоритмические основы мультимедийных технологий» и «Современные компьютерные технологии».

Студент первого курса магистратуры, приступая к изучению курса «Анализ алгоритмов и сложность вычислений», должен иметь базовую подготовку по курсам «Основы программирования», «Языки и методы программирования». Вместе с тем такие личностные характеристики как общая образованность, организованность и трудолюбие, самостоятельность, настойчивость в достижении цели необходимы при освоении дисциплины.

3. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОП магистратуры

Процесс изучения дисциплины направлен на формирование следующих элементов компетенций в соответствии с ФГОС ВО, ОП ВО и приобретения следующих знаний, умений, навыков и (или) опыта деятельности:

| Код компетенции | Формулировка компетенции | Перечень планируемых результатов обучения |
|---|--------------------------|---|
| Общепрофессиональные компетенции | | |
| ОПК-2 | | – |

| | | |
|--|---|--|
| | <p>ОПК-2.1. Применяет современные информационно-коммуникационные и интеллектуальные компьютерные технологии, инструментальные среды, программно-технические платформы для решения задач в области создания и применения искусственного интеллекта</p> <p>ОПК-2.2. Обосновывает выбор современных информационно-коммуникационных и интеллектуальных компьютерных технологий</p> <p>ОПК-2.3. Разрабатывает оригинальные программные средства, в том числе с использованием современных информационно-коммуникационных и интеллектуальных компьютерных технологий, для решения задач в области создания и применения искусственного интеллекта</p> | |
|--|---|--|

4. Объем, структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 3 зач. ед., 108 акад. час.

| № п/п | Темы (разделы) дисциплины, их содержание | Семестр | Виды учебных занятий, включая самостоятельную работу студентов, и их трудоемкость (в академических часах) | | | | | | Формы текущего контроля успеваемости Форма промежуточной аттестации (по семестрам) |
|-------|--|---------|---|--------------|--------------|--------------|-----------------------------|---------------------------|---|
| | | | Контактная работа | | | | | | |
| | | | лекции | практические | лабораторные | консультации | аттестационные испытания | самостоятельная работа | |
| 1 | Сложность алгоритмов | 1 | 2 | 3 | | | | 10 | |

| | | | | | | | | | |
|---|--|---|-----------|-----------|--|----------|------------|-------------|---------|
| 2 | Разделяй и властвуй | 1 | 2 | 3 | | 1 | | 10 | |
| 3 | Динамическое программирование | 1 | 2 | 3 | | 1 | | 10 | |
| 4 | Матроиды и жадные алгоритмы | 1 | 2 | 3 | | 1 | | 10 | |
| 5 | Линейное программирование | 1 | | 3 | | | | 10 | |
| 6 | Классы P и NP | 1 | 1 | 3 | | 1 | | 10 | |
| 7 | NP-полные задачи | 1 | 1 | 3 | | | | 7,7 | |
| 8 | PSPACE и задачи за пределами NP | | 1 | 3 | | | | | |
| | Всего за 1 семестр | | 12 | 24 | | 4 | 0,3 | 67,7 | Экзамен |
| | Всего | | 12 | 24 | | 4 | 0,3 | 67,7 | |

Содержание разделов дисциплины:

| |
|--|
| <p>Сложность алгоритмов Задача комбинаторной оптимизации. Определение алгоритма. Алгоритм полного перебора. Трудоемкость алгоритма. Трудоемкость в лучшем, худшем и среднем случаях. Алгоритм сортировки вставками. Анализ трудоёмкости. Асимптотическая верхняя, нижняя и средняя оценка сложности.</p> |
| <p>Разделяй и властвуй Метод «разделяй и властвуй». Алгоритм сортировки слиянием. Анализ трудоёмкости. Алгоритм быстрой сортировки. Анализ трудоёмкости. Деревья принятия решений. Нижняя оценка трудоёмкости сортировки. Алгоритм сортировки подсчётом. Алгоритм Карацубы быстрого умножения чисел. Алгоритм Штрассена рекурсивного умножения матриц. Основная теорема о рекуррентных соотношениях. Рекуррентное размещение бинарного дерева на квадратной решётке.</p> |
| <p>Динамическое программирование Метод динамического программирования. Свойство оптимальной подструктуры. Свойство перекрытия вспомогательных подзадач. Задача о наибольшей общей подпоследовательности. Алгоритм динамического программирования для задачи о наибольшей общей подпоследовательности. Алгоритм Флойда-Уоршелла для задачи обо всех кратчайших путях в графе. Задача о самом длинном простом пути и неприменимость динамического программирования.</p> |
| <p>Матроиды и жадные алгоритмы Концепция жадного алгоритма. Жадный алгоритм для задачи о размене монет. Условия применения жадного алгоритма. Алгоритм ближайшего соседа задачи коммивояжёра. Определение матроида. Примеры матроидов (матричный, графический, k-матроид и т.д.). Жадный алгоритм Радо-Эдмондса. Теорема Радо-Эдмондса. Примеры задач, которые не являются матроидами. Задача о планировании единичных заданий. Решение задачи о планировании единичных заданий с помощью матроида.</p> |

Линейное программирование

Постановка задачи линейного программирования. Выпуклый полиэдр и выпуклый многогранник. Основная теорема линейного программирования. Куб Кли-Минти и трудоёмкость симплекс-метода. Постановка задачи о максимальном потоке в форме линейного программирования. Теорема Форда-Фалкерсона. Постановка задачи о многопродуктовом потоке в форме линейного программирования. Постановка задачи о назначениях в форме линейного программирования. Целочисленное линейное программирование, релаксация линейного программирования. Вполне унимодулярные матрицы. Свойства полиэдра с вполне унимодулярной матрицей. Матрица инцидентности двудольного графа. Расширение многогранника. Сложность расширения. Многогранник и его расширение для задачи об остовном дереве. Примеры задач с экспоненциальной сложностью расширения.

Классы P и NP

Тезис Кобхема-Эдмондса. Полиномиальные и экспоненциальные алгоритмы. Задача распознавания. Детерминированная машина Тьюринга. Полиномиальный алгоритм. Класс задач P. Примеры задач из класса P. Недетерминированная машина Тьюринга. Машина Тьюринга с оракулом. Класс задач NP.

Примеры задач из класса NP. Класс задач co-NP. Дополнительные задачи. Задача соотношения классов P и NP.

NP-полные задачи

Полиномиальная сводимость. Полиномиальное сведение задачи о гамильтоновом цикле к задаче коммивояжера. NP-полная задача. NP-трудная задача. Сводимость по Тьюрингу. Задача выполнимости булевых формул. Теорема Кука-Левина. Схема доказательства NP-полноты задачи.

Доказательство методом сужения задачи. Основные NP-полные задачи (3-выполнимость, вершинное покрытие, клика и независимое множество, гамильтонов цикл и задача коммивояжер, сумма подмножества и задача о рюкзаке). Доказательство NP-полноты сужением задачи.

PSPACE и задачи за пределами NP

Класс задач PSPACE. Соотношение классов сложности P, NP, co-NP и PSPACE. Примеры PSPACE задач (задача планирования, выполнимость с кванторами, задача конкурентного размещения).

Алгоритмы для задач выполнимости с кванторами и конкурентного размещения с полиномиальным ограничением по памяти. PSPACE-полная задача

Доказательство PSPACE-полноты задачи о конкурентном размещении. Алгоритм минимакса и пошаговые игры для двух игроков.

5. Образовательные технологии, используемые при осуществлении образовательного процесса по дисциплине

В процессе обучения используются следующие образовательные технологии:

Вводная лекция – дает первое целостное представление о дисциплине и ориентирует студента в системе изучения данной дисциплины. Студенты знакомятся с назначением и задачами курса, его ролью и местом в системе учебных дисциплин и в системе подготовки в целом. Дается краткий обзор курса, история развития науки и практики, достижения в этой сфере, имена известных ученых, излагаются перспективные направления исследований. На этой лекции высказываются методические и организационные особенности работы в рамках данной дисциплины, а также дается анализ рекомендуемой учебно-методической литературы.

Академическая лекция (или лекция общего курса) – последовательное изложение материала, осуществляемое преимущественно в виде монолога преподавателя. Требования к академической лекции: современный научный уровень и насыщенная информативность, убедительная аргументация, доступная и понятная речь, четкая структура и логика, наличие ярких примеров, научных доказательств, обоснований, фактов.

Практическое занятие – занятие, посвященное освоению конкретных умений и навыков и закреплению полученных на лекции знаний.

Консультации – групповые занятия, на которых по просьбе студентов рассматриваются наиболее сложные моменты в решении задач, возникающие у них в процессе самостоятельной работы.

6 . Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного программного обеспечения и информационных справочных систем (при

необходимости)

В процессе осуществления образовательного процесса используются: для разработки документов, презентаций, для работы с электронными таблицами OfficeStd 2013 RUS OLP NL Acdmc 021-10232
LibreOffice (свободное)
издательская система LaTeX;
для поиска учебной литературы библиотеки ЯрГУ – Автоматизированная библиотечная информационная система "БУКИ-NEXT" (АБИС "Буки-Next")

7. Перечень основной и дополнительной учебной литературы, ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

а) основная:

- 1 Алгоритмы: построение и анализ: пер. с англ. / Т. Кормен, Ч. Лейзерсон, Р. Ривест, К. Штайн - 3-е изд. - М.; СПб.: Диалектика, 2020. - 1323 с.: ил.

Дополнительная литература:

- 2 Асанов, М. О. Дискретная математика: графы, матроиды, алгоритмы : учебное пособие / М. О. Асанов, В. А. Баранский, В. В. Расин. — 3-е изд., стер. — Санкт-Петербург : Лань, 2020. — 364 с.— Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/130477> (дата обращения: 06.10.2021). Режим доступа: для авториз. пользователей.
- 3 Методы оптимизации: теория и алгоритмы : учебное пособие для академического бакалавриата / А. А. Черняк, Ж. А. Черняк, Ю. М. Метельский, С. А. Богданович. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2018. — 357 с. — (Бакалавр. Академический курс). — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/415561> (дата обращения: 06.10.2021).
- 4 Крупский, В. Н. Теория алгоритмов. Введение в сложность вычислений : учебное пособие для бакалавриата и магистратуры / В. Н. Крупский. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2019. — 117 с. — (Авторский учебник) — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/444131> (дата обращения: 06.10.2021).

8. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине включает в свой состав специальные помещения:

- учебные аудитории для проведения занятий лекционного типа;
- учебные аудитории для проведения групповых и индивидуальных консультаций,
- учебные аудитории для проведения текущего контроля и промежуточной аттестации;
- помещения для самостоятельной работы;
- помещения для хранения и профилактического обслуживания технических средств обучения.

Специальные помещения укомплектованы средствами обучения, служащими для представления учебной информации большой аудитории.

Для проведения занятий лекционного типа предлагаются наборы учебно-наглядных пособий, хранящиеся на электронных носителях и обеспечивающие тематические иллюстрации, соответствующие рабочим программам дисциплин.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду организации.

Число посадочных мест в лекционной аудитории больше либо равно списочному составу потока, а в аудитории для практических занятий – списочному составу группы обучающихся.

Приложение №1 к рабочей программе дисциплины «Анализ алгоритмов и сложность вычислений»

Фонд оценочных средств для проведения текущей и промежуточной аттестации студентов по дисциплине

1. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы

формирования компетенций

1.1. Контрольные задания и иные материалы, используемые в процессе текущей аттестации

Задания для самостоятельной работы

Пример теста

1. Для двух функций $f(n)$ и $g(n)$, оценивающих трудоёмкость некоторого алгоритма, известно, что
 1. $\exists C > 0 \exists n_0 \forall n > n_0: f(n) \geq Cg(n)$,
 2. тогда
 - а) $f(n) = O(g(n))$;
 - б) $f(n) = o(g(n))$;
 - в) $f(n) = \Omega(g(n))$;
 - г) $f(n) = \Theta(g(n))$.
2. Решить рекуррентное соотношение
 3. $T(n) = 9T\left(\frac{n}{3}\right) + n^2$.
 - а) $\theta(n^3)$;
 - б) $\theta(n^2 \log n)$;
 - в) $\theta(n^2)$;
 - г) $\Theta(n \log n)$.
3. 1 000 000 студентов написали тест, который был оценен по 100-бальной шкале. Какой из приведённых алгоритмов позволит отсортировать результаты теста по возрастанию менее чем за 2 000 000 элементарных операций?
 - а) сортировка вставками;
 - б) быстрая сортировка;
 - в) пирамидальная сортировка;
 - г) сортировка подсчётом.
4. Сколько символов содержит самая длинная общая подпоследовательность последовательностей

4. $\langle 1,0,0,1,0,1,0,1 \rangle$ и $\langle 0,1,0,1,1,0,1,1,0 \rangle$

- а) 4;
 б) 5;
 в) 6;
 г) 7.
5. Что из перечисленного не является матроидом:
 а) множество клик в графе;
 б) множество лесов в графе;
 в) множество всех подмножеств конечно множества;
 г) множество линейно независимых строк матрицы?
6. Каждой вершине графа $G = (V, E)$ сопоставим переменную x_i . Какая задача записана в форме целочисленного линейного программирования:
 5. $\sum_{i=1}^{|V|} x_i \rightarrow \min,$
 6. $x_i + y_j \geq 1, \forall (i, j) \in E,$
 7. $x_i \in \{0,1\}.$
 а) задача о вершинном покрытии;
 б) задача о независимом множестве;
 в) задача о клике;
 г) задача коммивояжёра.
7. Найти оптимальное значение целевой функции в задаче линейного программирования
 8. $f(x, y) = 2x - y \rightarrow \max, 3x - 4y \leq 6,$
 $2x + 2y \leq 11,$
 9. $-4x + 3y \leq 6,$
 $x \geq 0,$
 $y \geq 0.$
 а) $f(x, y) = 3,5;$
 б) $f(x, y) = 5;$
 в) $f(x, y) = 6.5;$
 г) $f(x, y) = 8.$
8. Какая из перечисленных компонент не входит в структуру недетерминированной машины Тьюринга?
 а) бесконечная лента;
 б) читающая/пишущая головка;
 в) управляющее устройство;
 г) все вышеперечисленные компоненты входят в структуру недетерминированной машины Тьюринга.
9. Пусть задача A полиномиально сводится к известной NP-полной задаче B ($A \leq_p B$). Что можно сказать про задачу A ?
 а) задача A полиномиально разрешима;
 б) задача A является NP-полной;
 в) задача A является NP-трудной;
 г) ничего из вышеперечисленного.
10. Расположите классы сложности в верном порядке вложенности:
 а) $P \subseteq PSPACE \subseteq NP \subseteq EXPTIME;$
 б) $P \subseteq NP \subseteq PSPACE \subseteq EXPTIME;$ в) $PSPACE \subseteq P \subseteq NP \subseteq EXPTIME;$ г) $P \subseteq PSPACE \subseteq EXPTIME \subseteq NP.$

Правильные ответы

| Вопрос № | Правильный ответ | Вопрос № | Правильный ответ |
|----------|------------------|----------|------------------|
| 1 | в | 6 | а |
| 2 | б | 7 | в |
| 3 | г | 8 | г |
| 4 | в | 9 | г |
| 5 | а | 10 | б |

Критерии оценки

- «Отлично» – 9 правильных ответов;
- «Хорошо» – 7 правильных ответов;
- «Удовлетворительно» – 5 правильных ответов;
- «Неудовлетворительно» – 4 и менее правильных ответов.

Темы докладов

1. Пирамидальная сортировка. Анализ сложности
2. Выбор порядковой статистики с линейным ожидаемым временем
3. Поиск ближайшей пары точек методом «разделяй и властвуй»
4. Динамическое программирование для задачи о перемножении цепочки матриц
5. Динамическое программирование для задачи о расстоянии редактирования
6. Алгоритм Нидлмана – Вунша для выравнивания последовательностей
7. Динамическое программирование для задачи о триангуляции с минимальным весом
8. Динамическое программирование для задачи о независимом множестве в дереве
9. Код Хаффмана
10. Матроид трансверсалей
11. Симплекс-метод
12. Алгоритмы проталкивания предпотока
13. Алгоритм быстрой оболочки для двумерного и многомерного пространств
14. Венгерский алгоритм для задачи о назначениях
15. Полиномиальный алгоритм для задачи о китайском почтальоне
16. Алгоритм динамического программирования для предсказания вторичной структуры РНК
17. Алгоритм Гэйла - Шэпли построения устойчивого паросочетания
18. NP-полнота задачи о многопроцессорном расписании
19. Ориентированные деревья минимального веса. Многофазный жадный алгоритм
20. Алгоритм Штор-Вагнера для задачи о минимальном разрезе
21. Алгоритм Каргера для задачи о минимальном разрезе
22. Поточковый алгоритм планирования авиаперелетов
23. Поточковый алгоритм для задачи об выбывании бейсбольных команд из борьбы за чемпионство
24. Алгоритм динамического программирования для задачи синтаксического разбора
25. Алгоритм определения наличия пересекающихся отрезков
26. Субмодулярные функции и алгоритм Скрейвера
27. Алгоритм Эдмондса для пересечения матроидов
28. Алгоритм Карпа для задачи о цикле минимального среднего веса
29. Гридоиды и жадные алгоритмы
30. Полиматроиды и жадные алгоритмы
31. Ранцевая криптосистема Меркла — Хеллмана
32. Динамическое программирование в алгоритме Seam Carving для изменения размеров изображения с учётом содержания
33. Полиномиальный алгоритм для гиперболической версии «Сапёра»
34. Проектирование и маршрутизация телекоммуникационных сетей с помощью целочисленного линейного программирования
35. Решение головоломки «Shakashaka» с помощью целочисленного линейного программирования
36. Решение задачи разбиения на квадраты с помощью целочисленного линейного программирования
37. Решение головоломок «Судоку» и «Какуро» с помощью целочисленного линейного программирования
38. NP-полнота игры «The Legend of Zelda»
39. NP-полнота игры «Pokemon»

40. NP-полнота игры «Морской бой»
41. NP-полнота игры «Тетрис»
42. NP-полнота игры «Plumber»
43. NP-полнота игры «Сапер»
44. NP-полнота задачи о назначении целей для вооружения
45. NP-полнота игры «Змейка»
46. NP-полнота головоломки «Masyu»
47. NP-полнота головоломки «Shakashaka»
48. NP-полнота игры Portal с полями анти-экспроприации
49. NP-полнота головоломки «Нонограмма»
50. PSPACE-полнота игры в города для обычных и планарных графов
51. Динамическое программирование при подсчете результата матча в крикет
52. Экспоненциальная нижняя оценка на сложность задачи коммивояжера в форме линейного программирования
53. NP-полнота задачи о художественной галерее
54. NP-полнота игры «Лемминги»
55. Теорема Шефера о дихотомии. NP-полнота задачи 3-выполнимость при одном истинном литерале. NP-полнота задачи 3-выполнимость при различных литералах
56. Алгоритм динамического программирования для задачи Beat tracking (отслеживание такта в музыке)
57. NP-полнота головоломок «Судоку», «Какуро» и «Slitherlink»
58. NP-полнота задачи коммивояжера на евклидовой плоскости
59. NP-полнота задачи о дереве Штейнера на манхэттенской плоскости
60. NP-полнота вулканской головоломки Кал-тох
61. NP-полнота Match-3 головоломок
62. «Super Mario Bros.» NP и PSPACE полнота, полиномиальный алгоритм спасения принцессы
63. Играть в домино трудно, если играешь не один
64. NP-полнота игр «Threes!», «Fives», «1024!» и «2048»
65. NP-полнота головоломок в игре «The Witness»
66. NP-полнота оптимальной сборки кубика Рубика
67. PSPACE-полнота игры Sokoban
68. Подход к игре «Тантрикс» на основе целочисленного линейного программирования
69. PSPACE-полнота игры «Реверси»
70. Проходить через двери PSPACE-трудно

| Показатели | Критерии |
|--------------------------------------|---|
| Содержание доклада | Анализирует изученный материал, Выделяет наиболее значимые для раскрытия темы факты, научные положения, Соблюдает логическую последовательность в изложении материала |
| Аргументированно отвечает на вопросы | Проявляет критическое мышление |
| Представление доклада | Использует иллюстративные, наглядные материалы, |

Критерии оценки

- «Отлично» – доклад полностью соответствует описанным критериям;
- «Хорошо» – доклад соответствует описанным критериям за исключением некоторых замечаний не более чем по нескольким пунктам критериев;
- «Удовлетворительно» – доклад соответствует более чем половине описанных критериев;
- «Неудовлетворительно» – доклад не соответствует большей части описанных критериев.

Вариант билета на зачете

| Задания | Ответы |
|---|---|
| 1. Сформулировать определение матроида 10. | Раздел 4 |
| 2. Сформулировать постановку задачи 3-выполномость | Раздел 7 |
| 3. Сформулировать и доказать теорему о матрице инцидентности двудольного графа 11. | Раздел 5 |
| 4. Решить рекуррентное соотношение 12. $T(n) = 3T\left(\frac{n}{2}\right) + n \lg(n)$. | $T(n) = \Theta(n^{\log_2 3})$ |
| 5. Применим ли алгоритм динамического программирования к задаче о самом длинном простом пути в графе? | Не применим, так как задача о самом длинном пути не обладает свойством оптимальной подструктуры: самый длинный путь между вершинами A и B не является самым длинными путём между промежуточными вершинами на маршруте. |
| 6. Доказать, что следующая задача является NP-полной: Бонни и Клайд ограбили банк и вынесли n чеков на различные суммы, выписанные на их имя; требуется разделить чеки так, чтобы разница между долями Бонни и Клайда не превосходила 100 долларов. | Задача принадлежит классу NP, так как решение можно угадать и проверить за полиномиальное время (достаточно посчитать доли Бонни и Клайда и сравнить их) Проверку можно выполнить за время $O(n)$. Рассмотрим NP-полную задачу о разбиении множества. Задано множество целых неотрицательных чисел $A = \{a_1, a_2, \dots, a_n\}$. Найти такое подмножество $S \subset A$, что |

| | |
|--|---|
| | $\sum_{a_i \in S} a_i = \sum_{a_i \in A \setminus S} a_i.$ Построим частный случай задачи о Бонни и Клайде с весами: $b_i = 1000a_i.$ В исходном множестве существует разбиение тогда и только тогда, когда Бонни и Клайд могут разделить чеки с разницей не более 100 долларов. |
|--|---|

Критерии оценки

- «Отлично» – даны верные ответы на 5 вопросов из билета;
- «Хорошо» – даны верные ответы на 4 вопроса из билета;
- «Удовлетворительно» – даны верные ответы на 3 вопроса из билета;
- «Неудовлетворительно» – даны верные ответы на 2 и менее вопросов из билета.

2. Перечень компетенций, этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкалы оценивания

2.1. Шкала оценивания сформированности компетенций и ее описание

Оценивание уровня сформированности компетенций в процессе освоения дисциплины осуществляется по следующей трехуровневой шкале:

Пороговый уровень - предполагает отражение тех ожидаемых результатов, которые определяют минимальный набор знаний и (или) умений и (или) навыков, полученных студентом в результате освоения дисциплины. Пороговый уровень является обязательным уровнем для студента к моменту завершения им освоения данной дисциплины.

Продвинутый уровень - предполагает способность студента использовать знания, умения, навыки и (или) опыт деятельности, полученные при освоении дисциплины, для решения профессиональных задач. Продвинутый уровень превосходит пороговый уровень по нескольким существенным признакам.

Высокий уровень - предполагает способность студента использовать потенциал интегрированных знаний, умений, навыков и (или) опыта деятельности, полученных при освоении дисциплины, для творческого решения профессиональных задач и самостоятельного поиска новых подходов в их решении путем комбинирования и использования известных способов решения применительно к конкретным условиям. Высокий уровень превосходит пороговый уровень по всем существенным признакам.

2.2. Перечень компетенций, этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования

Шкала оценивания компетенций:

| Оценка в 100-балльной шкале | Оценка в 5-ти балльной шкале | Уровень сформированности компетенций |
|-----------------------------|----------------------------------|--------------------------------------|
| 0-54 баллов | неудовлетворительно (не зачтено) | недостаточный |
| 55-69 баллов | удовлетворительно (зачтено) | базовый |
| 70-85 баллов | хорошо (зачтено) | повышенный |
| 86-100 баллов | отлично (зачтено) | |

Критерии оценивания компетенций:

| Компетенции | Индикаторы достижения компетенций | Оценочные средства |
|-------------|---|--------------------|
| ОПК-2. | ОПК-2.1. применяет современные информационно-коммуникационные и интеллектуальные компьютерные технологии, инструментальные среды, программно-технические платформы для решений задач в области создания и применения искусственного интеллекта. | Пример теста |

| Компетенции | Индикаторы достижения компетенций | Оценочные средства |
|-------------|---|-------------------------|
| | ОПК-2.2. Обосновывает выбор современных информационно-коммуникационных и интеллектуальных компьютерных технологий | Темы докладов |
| | ОПК-2.3. Разрабатывает оригинальные программные средства, в том числе с использованием современных информационно-коммуникационных и интеллектуальных компьютерных технологий, для решения задач в области создания и применения искусственного интеллекта | Пример билета на зачёте |

3. Методические рекомендации преподавателю по процедуре оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Целью процедуры оценивания является определение степени овладения студентом ожидаемыми результатами обучения (знаниями, умениями, навыками и (или) опытом деятельности).

Процедура оценивания степени овладения студентом ожидаемыми результатами обучения осуществляется с помощью методических материалов, представленных в разделе

«Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций»

3.1 Критерии оценивания степени овладения знаниями, умениями, навыками и (или) опытом деятельности, определяющие уровни сформированности компетенций

Пороговый уровень (общие характеристики):

- владение основным объемом знаний по программе дисциплины;
- знание основной терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы без существенных ошибок;
- владение инструментарием дисциплины, умение его использовать в решении стандартных (типовых) задач;
- способность самостоятельно применять типовые решения в рамках рабочей программы дисциплины;
- усвоение основной литературы, рекомендованной рабочей программой дисциплины;
- знание базовых теорий, концепций и направлений по изучаемой дисциплине;
- самостоятельная работа на практических и лабораторных занятиях, периодическое участие в групповых обсуждениях, достаточный уровень культуры исполнения заданий.

Продвинутый уровень (общие характеристики):

- достаточно полные и систематизированные знания в объеме программы дисциплины;
- использование основной терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы, умение делать выводы;
- владение инструментарием дисциплины, умение его использовать в решении учебных и профессиональных задач;
- способность самостоятельно решать сложные задачи (проблемы) в рамках рабочей программы дисциплины;
- усвоение основной и дополнительной литературы, рекомендованной рабочей программой дисциплины;
- умение ориентироваться в базовых теориях, концепциях и направлениях по изучаемой дисциплине и давать им сравнительную оценку;
- самостоятельная работа на практических и лабораторных занятиях, участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

Высокий уровень (общие характеристики):

- систематизированные, глубокие и полные знания по всем разделам дисциплины;
- точное использование терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы, умение делать обоснованные выводы;

- безупречное владение инструментарием дисциплины, умение его использовать в постановке и решении научных и профессиональных задач;
- способность самостоятельно и творчески решать сложные задачи (проблемы) в рамках рабочей программы дисциплины;
- полное и глубокое усвоение основной и дополнительной литературы, рекомендованной рабочей программой дисциплины;
- умение ориентироваться в основных теориях, концепциях и направлениях по изучаемой дисциплине и давать им критическую оценку;
- активная самостоятельная работа на практических и лабораторных занятиях, творческое участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

3.2 Описание процедуры выставления оценки

В зависимости от уровня сформированности каждой компетенции по окончании освоения дисциплины студенту выставляется оценка «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Показатели и критерии, используемые при выставлении оценки подробно описаны в разделе «Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций».

Высокий уровень формирования компетенций соответствует оценке «отлично» за самостоятельные, контрольные работы и экзаменационную работу.

Продвинутый уровень формирования компетенций соответствует оценке «хорошо» за самостоятельные, контрольные работы и экзаменационную работу.

Пороговый уровень формирования компетенций соответствует оценке «удовлетворительно» за самостоятельные, контрольные работы и экзаменационную работу.

Оценка «отлично» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована на высоком уровне.

Оценка «хорошо» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована не ниже, чем на продвинутом уровне.

Оценка «удовлетворительно» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована не ниже, чем на пороговом уровне.

Оценка «неудовлетворительно» выставляется студенту, у которого хотя бы одна компетенция (полностью или частично формируемая данной дисциплиной) сформирована ниже, чем на пороговом уровне.

Оценка «зачет» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована не ниже, чем на пороговом уровне.

Оценка «незачтено» выставляется студенту, у которого хотя бы одна компетенция (полностью или частично формируемая данной дисциплиной) сформирована ниже, чем на пороговом уровне.

Приложение №2 к рабочей программе дисциплины «Анализ алгоритмов и сложность вычислений»

Методические указания для студентов по освоению дисциплины

Основной формой изложения учебного материала по дисциплине «Анализ алгоритмов и сложность вычислений» являются лекции, проводимые на основе презентаций, демонстрирующих студентам основные конструкции изучаемого языка. При этом в каждый момент времени студент видит на экране как исходный код документа, так и его визуальное представление, получаемое в результате компиляции исходного кода. После каждой лекции студенты получают просмотренную презентацию в электронной форме, а в процессе лекции записывают в свои конспекты только дополнительные комментарии преподавателя. С целью приобретения навыков и умений, по каждой теме студентам выдаются соответствующие задания для самостоятельной работы. На выполнение каждого задания отводится фиксированное время. Если сданная студентом работа содержит ошибки, то преподаватель указывает некоторые типичные для данной работы ошибки и дает студенту еще некоторое время на исправления. За каждый цикл исправления студенту начисляются штрафные очки, влияющие на итоговую оценку.

На экзамене проводится итоговая проверка приобретенных студентом знаний и умений. Студенту предлагается сверстать небольшой фрагмент математического текста и(или) иллюстрации, при этом студент должен аргументировать свой выбор тех или иных средств для решения данной задачи. Как было сказано выше, итоговая оценка выставляется с учетом штрафных очков, набранных за решение домашних заданий в течение семестра.

Учебно-методическое обеспечение самостоятельной работы студентов по дисциплине

В качестве учебно-методического обеспечения рекомендуется использовать литературу, указанную в разделе № 7 данной рабочей программы.