

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ ТАДЖИКИСТАН
МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«РОССИЙСКО-ТАДЖИКСКИЙ (СЛАВЯНСКИЙ) УНИВЕРСИТЕТ»

Естественнонаучный факультет

Кафедра «Информатики и ИТ»

«УТВЕРЖДАЮ»
«25» 00 2023 г.
Зав. кафедрой к.э.н., доцент
Лешукович А.И. Решетов

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по

Теория алгоритмов

09.03.03 «Прикладная информатика»

Душанбе 2023 г.

В результате освоения дисциплины «Теория алгоритмов» формируются следующие (общепрофессиональные, профессиональные) компетенции обучающегося

1) Общепрофессиональные компетенции выпускников и индикаторы их достижения

Код	Формируемая компетенция	Содержание этапа формирования компетенции	Форма контроля
ОПК-1	Способностью применять естественнонаучные и общепрофессиональные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности	<p>ИОПК-1.1. Применять основы математики, физики, вычислительной техники и программирования в профессиональной деятельности.</p> <p>ИОПК-1.2. Решать стандартные профессиональные задачи с применением естественнонаучных и общепрофессиональных знаний, методов математического анализа и моделирования.</p> <p>ИОПК-1.3. Использовать методы теоретического и экспериментального исследования объектов профессиональной деятельности.</p>	<p>Тестирование. Контроль самостоятельной работы. Отчеты по практическим работам. Контрольная работа. Устный опрос.</p>
ОПК-7	Способностью разрабатывать алгоритмы и программы, пригодные для практического применения	<p>ИОПК-7.1. Применять языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий.</p> <p>ИОПК-7.2. Применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ.</p> <p>ИОПК-7.3. Программировать, выполняет отладку и тестирование прототипов программно-технических ком-</p>	<p>Тестирование. Контроль самостоятельной работы. Отчеты по практическим работам. Контрольная работа. Устный опрос.</p>

		плексов задач.	
--	--	----------------	--

2) Профессиональные компетенции: проектная деятельность:

Код	Формируемая компетенция	Содержание этапа формирования компетенции	Форма контроля
ПК-2	Способностью разрабатывать и адаптировать прикладное программное обеспечение.	ИПК-2.1. Применять современные технологии разработки и адаптации прикладного программного обеспечения.	Тестирование. Контроль самостоятельной работы. Отчеты по практическим работам. Контрольная работа. Устный опрос.
		ИПК-2.2. Участвовать в разработке на современных языках программирования и адаптации прикладного программного обеспечения	
		ИПК-2.3. Применять современные технологии для разработки веб-приложений	

**ПАСПОРТ
ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ**

по дисциплине **Теория алгоритмов**

№ п/п	Контролируемые разделы, темы, модули ¹	Формируемые компетенции	Оценочные средства		
			Количество тестовых заданий	Другие оценочные средства	Вид
1	Примеры алгоритмов. Основные свойства интуитивного понятия алгоритма. Числовые функции: частичные,totальных. Понятие интуитивно вычислимой функции и разрешимого множества. Необходимость математических моделей алгоритмов. Основные типы моделей алгоритмов.	ОПК-1	6	Реферат	1
2	Матины Тьюринга как математическая модель алгоритма. Тезис Тьюринга. Вычисление функций на машинах Тьюринга. Построение машин Тьюринга. Тезисы Черча. Машина Поста. Вычисление функций на машинах Поста. Построение машин Поста.	ОПК-7	8	Письменная работа	1
3	Базисные функции: нулевая, следования, проекции. Операторы	ПК-2	10	Контрольная работа	1

	ры суперпозиции и примитивной рекурсии. Примитивно-рекурсивные функции. Оператор минимизации. Частично-рекурсивные функции. Тотально-рекурсивные функции. Примеры примитивно (частично, totally)-рекурсивных функций. Тезис Черча.				
4	Нормальные алгоритмы Маркова как математическая модель алгоритма. Принцип нормализации Маркова. Вычисление функций нормальными алгоритмами. Доказательство равнообъемности математических моделей алгоритмов: машин Тьюринга, частично-рекурсивных функций.	ОПК-7	18	Контрольная работа	1
5	Характеристическая функция множества. Определение рекурсивных и перечислимых множеств. Перечислимость рекурсивных множеств. Критерий рекурсивности.	ОПК-1	12	Контрольная работа	1
6	Кодирование машин Тьюринга, универсальная машина Тьюринга. Перечислимость множества частично-рекурсивных функций. Универсальная частично-рекурсивная функция. Существование универсальной функции для множества n -местных частично-рекурсивных функций.	ПК-2	8	Контрольная работа	1
7	Массовые алгоритмические проблемы. Неразрешимость проблемы остановки машин Тьюринга. Алгоритмическая сводимость. Обзор алгоритмически неразрешимых проблем	ОПК-1	12	Контрольная работа	1
8	Сортировка и определение сложности алгоритмов сортировки. Сортировка вставками. Пузырьковая сортировка. Сортировка выбором. Быстрая сортировка. Сортировка слиянием. Пирамидальная сортировка. Сортировка перечислением. Сортировка всплытием. Сортировка бинарным поиском. Алгоритмы сортировки, использующие структуру элементов: цифровая сортировка, корневая сортировка	ПК-2	8	Контрольная работа	1

ТЕМЫ РЕФЕРАТОВ И ПИСЬМЕННЫХ РАБОТ

(рефератов, эссе, письменных работ)

1. Примеры алгоритмов. Основные свойства интуитивного понятия алгоритма.
2. Необходимость математических моделей алгоритмов.
3. Основные типы моделей алгоритмов.
4. Машины Тьюринга как математическая модель алгоритма.
5. Тезис Тьюринга. Вычисление функций на машинах Тьюринга.
6. Машина Поста. Вычисление функций на машинах Поста.
7. Базисные функции: нулевая, следования, проекции.
8. Операторы суперпозиции и примитивной рекурсии.
9. Примитивно-рекурсивные функции.
10. Оператор минимизации. Частично-рекурсивные функции. Тотально-рекурсивные функции.
11. Примеры примитивно (частично, totally)-рекурсивных функций. Тезис Черча
12. Вычисление функций нормальными алгорифмами.
13. Характеристическая функция множества.
14. Кодирование машин Тьюринга: универсальная машина Тьюринга.
15. Массовые алгоритмические проблемы.
16. Неразрешимость проблемы остановки машин Тьюринга.
17. Сортировка и определение сложности алгоритмов сортировки.

Критерии оценки выполнения самостоятельной работы.

В основу разработки балльно рейтинговой системы положены принципы, в соответствии с которыми формирование рейтинга студента осуществляется постоянно в процессе его обучения в университете. Настоящая система оценки успеваемости студентов основана на использовании совокупности контрольных точек, равномерно расположенных на всем временном интервале изучения дисциплины. При этом предполагается разделение всего курса на ряд более или менее самостоятельных, логически завершенных блоков и модулей и проведение по ним промежуточного контроля.

Студентам выставляются следующие баллы за выполнение задания к ПК:

- оценка «отлично» (10 баллов): контрольные тесты, а также самостоятельно выполненные семестровые задания, выполненные полностью и сданные в срок в соответствии с предъявляемыми требованиями;
- оценка «хорошо» (8-9 баллов): задание выполнено и в целом отвечает предъявляемым требованиям, но имеются отдельные замечания в его оформлении или сроке сдачи;
- оценка «удовлетворительно» (6-7 баллов): задание выполнено не до конца, отсутствуют ответы на отдельные вопросы, имеются отклонения в объеме, содержании, сроке выполнения;
- оценка «неудовлетворительно» (5 и ниже): отсутствует решение задачи, задание переписано (сканано) из других источников, не проявлена самостоятельность при его выполнении.

Текущий контроль осуществляется в ходе учебного процесса по результатам выполнения самостоятельной работы и контрольной работы.

Основными формами текущего контроля знаний являются:

- обсуждение вынесенных в планах практических занятий лекционного материала и контрольных вопросов;
- решение тестов и их обсуждение с точки зрения умения сформулировать выводы, вносить рекомендации и принимать адекватные управленческие решения;
- выполнение контрольной работы и обсуждение результатов;
- участие в дискуссиях в качестве участника и модератора групповой дискуссии по темам дисциплины;
- написание и презентация доклада;
- написание самостоятельной (контрольной) работы.

Для контроля усвоения данной дисциплины учебным планом предусмотрен экзамен. Общее количество баллов по дисциплине - 100 баллов. Распределение баллов на текущий и промежуточный контроль при освоении дисциплины, а также итоговой оценке представлено ниже.

КОНТРОЛЬНЫЕ ЗАДАНИЯ И ВОПРОСЫ ДЛЯ ТЕКУЩЕГО

КОНТРОЛЯ ЗНАНИЙ ПО ДИСЦИПЛИНЕ

(ДЛЯ ТЕКУЩЕЙ АТТЕСТАЦИИ И КОНТРОЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ)

18. Примеры алгоритмов. Основные свойства интуитивного понятия алгоритма.
19. Числовые функции: частичные, тотальные.
20. Понятие интуитивно вычислимой функции и разрешимого множества.

21. Необходимость математических моделей алгоритмов.
22. Основные типы моделей алгоритмов.
23. Машины Тьюринга как математическая модель алгоритма.
24. Тезис Тьюринга. Вычисления функций на машинах Тьюринга.
25. Построение машин Тьюринга. Тезисы Черча.
26. Машина Поста. Вычисление функций на машинах Поста.
27. Построение машин Поста.
28. Базисные функции: нулевая, следования, проекции.
29. Операторы суперпозиции и примитивной рекурсии.
30. Примитивно-рекурсивные функции.
31. Оператор минимизации. Частично-рекурсивные функции. Тотально-рекурсивные функции.
32. Примеры примитивно (частично, totally)-рекурсивных функций. Тезис Черча
33. Нормальные алгоритмы Маркова как математическая модель алгоритма.
34. Принцип нормализации Маркова.
35. Вычисление функций нормальными алгоритмами.
36. Доказательство равнобъемности математических моделей алгоритмов: машин Тьюринга, частично-рекурсивных функций
37. Характеристическая функция множества. Определение рекурсивных и перечислимых множеств. Перечислимость рекурсивных множеств. Критерий рекурсивности
38. Кодирование машин Тьюринга. Универсальная машина Тьюринга.
39. Перечислимость множества частично-рекурсивных функций.
40. Универсальная частично-рекурсивная функция.
41. Существование универсальной функции для множества n -местных частично-рекурсивных функций
42. Массовые алгоритмические проблемы.
43. Неразрешимость проблемы остановки машин Тьюринга.
44. Алгоритмическая сводимость
45. Обзор алгоритмически неразрешимых проблем
46. Сортировки и определение сложности алгоритмов сортировки.
47. Сортировки вставками. Пузырьковая сортировка.
48. Сортировки выбором. Быстрая сортировка. Сортировка слиянием.
49. Пирамидальная сортировка. Сортировка перечислением.
50. Сортировка всплытием. Сортировка бинарным поиском.
51. Алгоритмы сортировки, использующие структуру элементов: цифровая сортировка, корневая сортировка

Тесты по теории алгоритмов

(@1. Электронно-вычислительная машина (ЭВМ) – это

\$A) Комплекс технического оборудования, который способен выполнять любые указания человека; **\$B)** Многофункциональное техническое устройство, позволяющее вводить, обрабатывать и выводить информацию, решая различные прикладные задачи; **\$C)** Электронное устройство, которое выполняет операции ввода информации, хранения и обработки ее по определенной программе; **\$D)** Устройство, которое так же как и человек, получает информацию, хранит и обрабатывает ее, обменивается ею с другими компьютерами; **\$E)** Устройство, способное самостоятельно принимать данные, выполнять над ними указанные операции;

(@2. Информация – это

\$A) Любые сведения, которые интересуют конкретного человека в конкретной ситуации; **\$B)** Сведения об объектах и явлениях окружающей среды, их параметрах, свойствах и состояниях, которые уменьшают имеющуюся о них степень неопределенности методические; **\$D)** Указания, рекомендации, статьи, рефераты докладов, документы и т.п.; **\$C)** Сведения о лицах, предметах, фактах, событиях, явлениях и процессах независимо от формы их представления; **\$E)** Совокупность данных, повышающих уровень знаний об объективной реальности окружающего мира;

(@3. Основными особенностями алгоритма - это

\$A) Определенность, ввод, вывод, детерминированность; **\$B)** Явность, сходность, детерминированность, вывод; **\$C)** Определенность, понятность, сходность, ввод, вывод, детерминированность; **\$D)** Явность, сходность, ввод, вывод, детерминированность; **\$E)** Активность, ввод, вывод, детерминированность, сходность;

(@4. К свойствам информации относятся следующие

\$A) Объективность; **\$B)** Объемность; **\$C)** Необходимость; **\$D)** Полнота; **\$E)** Субъективность;

@5. Информацию измеряют

- \$A) Количество новизны; \$B) Числовой характеристикой сигнала, характеризующую неопределенность, которая исчезает после получения сообщения в виде данного сигнала; \$C) Количество символов в сообщении; \$D) Обыкновенным голосованием; \$E) Уменьшении неопределенности наших знаний об объекте;

@6. К свойствам информации относится следующие

- \$A) Объективность; \$B) Достоверность; \$C) Необходимость; \$D) Объемность; \$E) Субъективность;

@7. Термин «информация» происходит от латинского *informatio*, что означает

- \$A) Изложение, разъяснение; \$B) Сведения, информация; \$C) События, изложение; \$D) Разъяснение, обсуждение; \$E) Свойства, характеристика;

@8. Детерминированность алгоритма – это

- \$A) Последовательное выполнение блоков алгоритма; \$B) Порядок выполнения действий при решении задач; \$C) После выполнения очередного шага алгоритма однозначно определено, что делать на следующем шаге; \$D) Порядок размещения операнд в выражение; \$E) Последовательность использования арифметических операций в алгоритме;

@9. Алгоритм разбивается на отдельные шаги (этапы), каждый из которых должен быть простым и локальным. Эта особенность алгоритма является

- \$A) Детерминированность; \$B) Компактность; \$C) Универсальность; \$D) Определенность; \$E) Условность;

@10. Примерами «почти» алгоритмов являются

- \$A) Решение задач, определение корней уравнений; \$B) Медицинский рецепт, правила дорожного движения; \$C) Компьютерная программа. Кулинарный рецепт; \$D) Медицинский и кулинарный рецепты; \$E) Порядок решения задач, нахождение области определения функции;

@11. Роль исходных данных для алгоритма играют

- \$A) Данные; \$B) Параметры; \$C) Операнды; \$D) Переменные; \$E) Файлы;

@12. Идентификатор – это

- \$A) Имя архивного файла; \$B) Область видимости программы; \$C) Имя программного объекта; \$D) Программа, тестирующая правильности работы процессора; \$E) Тестирующие программы;

@13. Архитектура — это

- \$A) Общие принципы построения ЭВМ, реализующие программное управление работой и взаимодействием основных ее функциональных узлов; \$B) Общие принципы построения ЭВМ, не реализующие программное управление работой; \$C) Дизайн внешнего вида ЭВМ; \$D) Принцип соединения внешних устройств к ЭВМ; \$E) Совокупность основных устройств вычислительных машин;

@14. Центральный процессор выполняет следующие функции

- \$A) Выполняет арифметические и логические операции; \$B) Управляет процессами приема, обработки, хранения и передачи данных; \$C) Осуществляет физическое управление устройствами; \$D) Контролирует состояние устройств; \$E) Хранит активные программы и данные;

@15. Оперативная память компьютера – это

- \$A) Память, предназначенная для временного хранения данных и команд, необходимых для передачи информации; \$B) Маленькие схемы памяти, которые вставляются в материнскую плату; \$C) Микросхема с записанным набором программ; \$D) Место длительного хранения данных; \$E) Участок памяти для хранения информации, который исчезает при выключении питания ЭВМ;

@16. Оперативная память предназначена для

- \$A) Выполнения арифметических и логических операций; \$B) Управления процессами передачи данных; \$C) Физического управления устройствами; \$D) Хранения активных программ и данных хранения активных программ и данных; \$E) Контроля состояния устройств;

@17. Основными характеристиками памяти являются

- \$A) Емкость; \$B) Быстродействие; \$C) Разрядность; \$D) Тактовая частота; \$E) Способность обеспечивать работы всех устройств ЭВМ;

@18. КЭШ память - это

- \$A) Устройство для повышения производительности процессора; \$B) Память для временного хранения наиболее часто используемых программ; \$C) Сверхбыстродействующая память, являющаяся промежуточной между основной памятью и процессором; \$D) Периферийное устройство для хранения данных и про-

грамм выполняемых в данный момент; \$E) Функциональная возможность компьютера, чтобы закачивать туда игры, музыку, фильмы;

@19. Основная идея получить все вычислимые функции из существенно ограниченного множества базисных функций с помощью простейших алгоритмических средств принадлежит
\$A) Тьюрингу; \$B) Стивену Коулоу Клини; \$C) Гёлделю; \$D) Чёрчу; \$E) Эбрану;

@20. Подход к формализации понятия алгоритма принадлежит
\$A) Гёлделю и Клини; \$B) Тьюрингу; \$C) Клини; \$D) Эбрану; \$E) Гёлделю;

@21. Вычисления, заданные программой, реализуются
\$A) Устройством ввода алфавитно-цифровой и графической информации; \$B) Устройство хранения данных с произвольным доступом; \$C) Устройство хранения данных на смснных лазерных дисках; \$D) Устройством обработки табличных данных; \$E) Центральным процессором;

@22. Средства связи между устройствами и блоками машины, а также между машиной и пользователем, называются
\$A) Локальная сеть; \$B) Интерфейс; \$C) комплекс терминалов, подключенных каналами связи к большой ЭВМ; \$D) мультимедийный компьютер с принтером, модемом и факсом; \$E) Устройство связи;

@23. Виртуальная память - это
\$A) Временный участок памяти для хранения вводимых информации; \$B) Временный участок памяти для хранения результатов обработки информации; \$C) Комплекс объединенных запоминающих устройств для совместного решения задач; \$D) Совокупность внутренних и внешних памятий; \$E) Расширение адресного пространства задач за счет использования внешней памяти;

@24. Все функции, которые можно получить из базисных функций за конечное число шагов только с помощью определенных механизмов, называются
\$A) Примитивно-рекурсивными; \$B) Общерекурсивными; \$C) Частично-рекурсивными; \$D) Офисных технологий; \$E) Вычислимыми;

@25. Если функция получается всюду определенная, то она называется
\$A) Общерекурсивной; \$B) Вычислимой; \$C) Частично-рекурсивной; \$D) Примитивно-рекурсивной; \$E) Рекурсивной;

@26. Для организации виртуальной памяти на жестком диске создается файл подкачки и этот файл занимает
\$A) Половины оперативно памяти; \$B) 25 % кэш памяти; \$C) Место в постоянной памяти; \$D) 10 % емкости жесткого диска; \$E) Верхний регистр устройства ввода;

@27. В системное программное обеспечение входят
\$A) Языки программирования; \$B) Операционные системы; \$C) Графические редакторы; \$D) Компьютерные игры; \$E) Текстовые редакторы;

@28. В прикладное программное обеспечение входят
\$A) языки программирования; \$B) операционные системы; \$C) диалоговая оболочка; \$D) совокупность всех программ, установленных на компьютере; \$E) текстовые редакторы;

@29. Зарезервированные идентификаторы, которые имеют специальное значение для компилятора, называются
\$A) Операнд; \$B) Ключевые слова; \$C) Устройство ввода; \$D) Лексемы; \$E) Устройство, определяющее порядок выполнения операций в программе;

@30. Программа - это
\$A) Текст, оформленный по определенным правилам; \$B) Описание алгоритма на языке, понятном исполнителю; \$C) Алгоритм, записанный на языке программирования; \$D) Документ, излагающий основные направления работы учреждения; \$E) План действия, деятельности, работы с компьютером;

@31. Алгоритм - это
\$A) Описание последовательности действий для решения задачи или достижения поставленной цели; \$B) Правила выполнения основных операций обработки данных; \$C) Описание вычислений по математическим формулам; \$D) Множество способов решения поставленной задачи, связанных воедино; \$E) Точное предпи-

сание исполнителю совершить определенную последовательность действий для достижения поставленной цели за конечное число шагов;

@32. Свойство алгоритма, обеспечивающее решение не одной задачи, а целого класса задач этого типа

\$A) понятность; \$B) определенность; \$C) дискретность; \$D) массовость; \$E) результативность;

@33. Свойство алгоритма, обеспечивающее прекращение за конечное число шагов с определенным ответом на поставленную задачу

\$A) Понятность; \$B) Детерминированность; \$C) Дискретность; \$D) Результативность; \$E) Эффективность;

@34. Процесс создания программ с использованием различных языков программирования - это

\$A) Алгоритм решения задачи; \$B) Программирование; \$C) Отладка программы; \$D) Язык программирования; \$E) Обработка данных;

@35. Как называется графическое представление алгоритма

\$A) Блок-схема; \$B) Последовательность формул для решения задач; \$C) Табличное представление информации; \$D) Словесное описание действий пользователя; \$E) Последовательность инструкций для выполнения определенных действий;

@36. Совокупность программ, позволяющих осуществлять на компьютере автоматизированную обработку информации - это

\$A) Прикладные программы; \$B) Системные программы \$C) Пакеты прикладных программ; \$D) Операционные системы; \$E) Программное обеспечение;

@37. Язык программирования - это

\$A) Набор слов для написания программы; \$B) Определенная последовательность бит; \$C) Формализованные языки для создания программ, используемых на ЭВМ; \$D) Средство общения между человеком и компьютером; \$E) Совокупность правил образования и истолкования конструкций из символов для задания алгоритмов;

@38. Если функция получена без механизма минимизации, то она называется

\$A) Рекурсивной; \$B) Примитивно-рекурсивной; \$C) Частично-примитивной; \$D) Частично-рекурсивной; \$E) Вычислимой;

@39. Алгоритмы типичным образом решают не только частные задачи, но решают

\$A) Классы задач; \$B) Общие задачи; \$C) Универсальные задачи; \$D) математические задачи; \$E) Программируемые задачи;

@40. Поименованная последовательность символов, стандартная структура которой обеспечивает ее размещение в памяти ЭВМ - это

\$A) Операнд; \$B) Имена переменных; \$C) Папка; \$D) Файл; \$E) Программа;

@41. Языками программирования являются следующие

\$A) Excel, WordPad, Paint; \$B) Word, Excel, Pascal; \$C) Access, Excel, Pascal; \$D) Basic, Paint, Supercalc; \$E) Basic, Pascal, Fortran;

@42. Режим Plug-and-play обеспечивает

\$A) Автоматизированную установку языков программирования; \$B) Установку антивирусных программ; \$C) Установку и настройку программ без вмешательства пользователя; \$D) Обновлению операционной системы; \$E) Созданию копии файлов;

@43. Подлежащие решению частные задачи, выделяемые по мере надобности из рассматриваемого класса, определяются с помощью

\$A) Исходных данных; \$B) Параметров; \$C) Промежуточных результатов; \$D) результатов; \$E) Значений;

@44. Если для k -местной функции $f: \mathbb{N}^k \rightarrow \mathbb{N}$ существует вычисляющий алгоритм A, то он называется

\$A) Вычислимой функцией; \$B) Рекурсивной функцией; \$C) Частично-рекурсивной функцией; \$D) Примитивно-рекурсивной функцией; \$E) Общерекурсивной функцией;

@45. Если на вход алгоритма A поступил вектор $X = \langle x_1, x_2, \dots, x_k \rangle$ из области определения функции $f(x)$, то вычисления должно

§A) На выход появляется функция $f(x)$; §B) Повторяется функция $f(x)$; §C) На выход появляется рекурсивная функция; §D) Закончиться после конечного числа шагов и выдать функцию $f(x)$; §E) Заканчиваться после вывода функции $f(x)$;

@46. Если на вход алгоритма A поступил вектор $X = \langle x_1, x_2, \dots, x_k \rangle$ не принадлежащий области определения функции $f(x)$, то

§A) Алгоритм A несколько раз повторяется; §B) Исходная функция выводится на экран; §C) Алгоритм A заканчивается после конечных шагов цикла; §D) Исходная функция вычисляется конечным числом шагов алгоритма; §E) Алгоритм A никогда не заканчивается;

@47. Операционная система - это

§A) Основный блок компьютера; §B) Программа, выполняющая арифметические и логические операции; §C) Система программ, осуществляющая общее управление работой устройств компьютера; §D) Программа, управляющая работой компьютера в каждый конкретный момент времени; §E) Программа, обеспечивающая доступ пользователя к ресурсам компьютера;

@48. В состав операционной системы входит следующая подсистема:

§A) Управления устройствами; §B) Файловая система; §C) Управления базами данных; §D) Управления памятью компьютера; §E) Управления программами;

@49. Механизмы Гёделя являются следующие алгоритмические способы:

§A) Оператор суперпозиция, примитивная рекурсия, оператор минимизация; §B) Оператор перестановки, примитивная рекурсия, оператор минимизация; §C) Оператор экстремума, общая рекурсия, оператор перестановки; §D) Оператор суперпозиция, примитивная рекурсия, оператор локализация; §E) Операция суперпозиции, частичная рекурсия, операция минимизации;

@50. Оператор суперпозиции и примитивная рекурсия позволяют задать:

§A) Функцию, которая принадлежит области определения оператора; §B) Только четных функций; §C) Только нечетных функций; §D) Только всюду определенные функции; §E) Примитивную функцию;

@51. Пусть N - множество натуральных чисел, D_f - область определения функции $f(x)$, R_f - область значений функции $f(x)$. Если объекты которые рассматриваются будут функциями с областью определения D_f не принадлежащей N^k и с областью значений R_f не принадлежащей N , то такие функции называются

§A) Рекурсивными функциями; §B) Частично-рекурсивными функциями; §C) Общерекурсивными функциями; §D) Вычислимными функциями; §E) k -местными частичными функциями;

@52. К прикладному программному обеспечению относятся

§A) Новые языки программирования и компиляторы к ним, интерфейсные системы; §B) Решение вопросов об анализе потоков информации в различных сложных системах; §C) Системы обработки текстов, электронные процессоры, базы данных; §D) Поисковые системы, глобальные системы хранения и поиска информации; §E) Системы обработки табличных данных;

@53. Множество исходных функций обладают следующими свойствами:

§A) Постоянная функция $0(x) = 1$, одноместная функция следования $s(x) = x$, функция проекции $pr_i(x) = x_i + 1$; §B) Постоянная функция $0(x) = 0$, одноместная функция следования $s(x) = x + 1$, функция проекции $pr_i(x) = x_i$; §C) Постоянная функция $p(x) = 0$, одноместная функция ветвления $s(x) = x + 1$, функция оператора $pr_i(x) = x_i$; §D) Циклическая функция $O(x) = c$, рекурсивная функция $s(x) = x + p$, функция проекции $pr_i(x) = x_i$; §E) Постоянная функция $C(x) = C$, одноместная функция $s(x) = x + C$, функция проекции $pr_i(x) = x_i + p$ рабочего стола;

@54. Если k -местная функция получена из m -местной функции $\phi(y_1, y_2, \dots, y_m)$ и k -местных функций $f(x) = \phi(g_1(x), g_2(x), \dots, g_m(x))$, то говорят, что он получена с помощью

§A) Оператора суперпозиции; §B) Операции рекурсии; §C) Оператора минимизации; §D) Оператора рекурсии; §E) Программы минимизации;

@55. Операция, которая ставит в соответствие частичной функции $f: N^{k+1} \rightarrow N$ частичную функцию $h: N^k \rightarrow N$, и определяется так: область определения D_h существует, $h(x) = \min\{y\}$ при котором $f(x, y) = 0$ – это

§A) Оператор рекурсии; §B) Операция отражения; §C) Операция преобразования; §D) Оператор минимизации; §E) Оператор суперпозиции;

@56. Оператор минимизации обозначается следующим образом:

§A) $f(x) = \mu x [f(x, y) = 0]$; §B) $h(x) = \mu y [f(x, y) = 1]$; §C) $h(x) = \mu y [f(x, y) - 1 = 0]$; §D) $h(x) = \mu y [f(x + 1, y) = 0]$; §E) $h(x) = \mu y [f(x, y) = 0]$;

@57. Частично-рекурсивная функция sum: $\langle x, y \rangle \rightarrow x + y$ означает

§A) Сложение двух рекурсивных функций; §B) Объединение двух операций; §C) Сложение двух чисел; §D) Сложение двух частичных функций; §E) Отражение рекурсивных функций;

@58. Частично-рекурсивная функция prod: $\langle x, y \rangle \rightarrow xy$ означает

§A) Умножение двух рекурсивных функций; §B) Умножение двух чисел; §C) Умножение двух четных чисел; §D) Умножение двух частичных функций; §E) Отражение рекурсивных функций;

@59. Частично-рекурсивная функция $\delta(x) = x - 1$ ($x > 0$), $\delta(x) = 0$ означает

§A) Усеченное вычитание 1; §B) Вычитание двух чисел; §C) Операцию вычитания; §D) Оператор уменьшение; §E) Оператор минимизации;

@60. Частично-рекурсивная функция $|x - y| = x - y$ ($x \geq y$), $|x - y| = y - x$ ($x < y$) означает

§A) Вычитание двух чисел; §B) Вычитание двух операторов; §C) Разность двух операторов; §D) Модуль разности; §E) Модуль операции вычитания двух функций;

@61. Частично-рекурсивная функция $x + y = x - y$ ($x \geq y$), $x + y = 0$ ($x < y$) означает

§A) Модуль разности; §B) Вычитание двух операторов; §C) Модуль операции вычитания; §D) Разность двух операторов; §E) Усеченная разность;

@62. Частично-рекурсивная функция $\min\{x, y\}$ означает

§A) Минимизация двух чисел; §B) Минимизация двух операторов; §C) Наименьшее из чисел x и y ; §D) Минимизация рекурсивных функций; §E) Наименьшее из двух частично-рекурсивных чисел;

@63. Существуют следующие виды констант:

§A) Целые, вещественные, символьные, строковые; §B) Целые, вещественные, комплексные, символьные; §C) Целые, вещественные, комплексные, строковые; §D) Данные, абсолютные, символьные, строковые; §E) Данные, целые, вещественные, символьные, строковые;

@64. Частично-рекурсивная функция $0! = 1$, $(y + 1)! = \text{prod}(y!, y + 1)$ означает

§A) Факториал заданного оператора; §B) Факториал заданного числа; §C) Факториал рекурсивной функции; §D) Увеличение двух чисел на 1; §E) Факториал заданного числа шагом 1;

@65. Если $g(x_1, x_2)$ – примитивно-рекурсивная функция и $f(x_1, x_2, x_3) = g(x_1, x_2)$, то $f(x_1, x_2, x_3)$ является

§A) Примитивно-рекурсивной функцией; §B) Частично-рекурсивной функцией; §C) Общерекурсивной функцией; §D) Нечетной функцией; §E) Трехмерной функцией;

@66. Если $g(x_1, x_2)$ – примитивно-рекурсивная функция и $f(x_1, x_2, x_3) = g(x_1, x_2)$, то $f(x_1, x_2, x_3)$ является примитивно-рекурсивной функцией и эта операция называется

§A) Минимизация; §B) Перестановка переменных; §C) Преобразование переменных; §D) Введение фиктивных переменных; §E) Минимизация переменных;

@67. Если $g(x_1, x_2)$ – примитивно-рекурсивная функция и $f(x_2, x_1) = g(x_1, x_2)$, то $f(x_1, x_2)$ является

§A) Общерекурсивной функцией; §B) Частично-рекурсивной функцией; §C) Рекурсивной функцией; §D) Нечетной функцией; §E) Примитивно-рекурсивной функцией;

@68. Если $g(x_1, x_2)$ – примитивно-рекурсивная функция и $f(x_2, x_1) = g(x_1, x_2)$, то $f(x_1, x_2)$ является также примитивно-рекурсивной функцией и эта операция называется

§A) Преобразование переменных; §B) Введение новых переменных; §C) Перестановка переменных; §D) Отождествление переменных; §E) Введение фиктивных переменных;

@69. Если $g(x_1, x_2, x_3)$ – примитивно-рекурсивная функция и $f(x_2, x_1, x_3) = g(x_1, x_2, x_3)$, то $f(x_1, x_2)$ является

§A) Рекурсивной функцией; §B) Примитивно-рекурсивной функцией; §C) Четной функцией; §D) Нечетной функцией; §E) Частично-рекурсивной функцией;

@70. Если $g(x_1, x_2, x_3)$ – примитивно-рекурсивная функция и $f(x_2, x_1, x_3) = g(x_1, x_2, x_3)$, то $f(x_1, x_2)$ является также примитивно-рекурсивной функцией и эта операция называется

§A) Отождествление переменных; §B) Перестановка переменных; §C) Введение фиктивных переменных; §D) Преобразование переменных; §E) Размещение переменных;

@71. Любую примитивно-рекурсивную функцию можно вычислить с помощью

§A) Цикла в форме *while*; §B) Условие в форме *if*; §C) Цикла в форме *for next*; §D) Цикла в форме *for*; §E) Цикла в форме *do while*;

@72. Оператор минимизации позволяет описать функции, которые нельзя вычислить за заранее ограниченное число итераций, для вычисления их значений требуется

§A) Цикла в форме *while*; §B) Условие в форме *if*; §C) Цикла в форме *for next*; §D) Цикла в форме *for*; §E) Цикла в форме *do while*;

@73. Машина Тьюринга предназначена для

§A) Преобразования вычислимых функций; §B) Определения рекурсивных функций; §C) Преобразования рекурсивных функций; §D) Выполнения арифметических операций; §E) Определения вычислимых функций;

@74. На неформальном уровне машину Тьюринга можно описывать как

§A) Микрокалькулятор; §B) Абак; §C) Вычислительная машина без устройства памяти; §D) Черный ящик с лентой; §E) Диск с лентой;

@75. Машине Тьюринга выглядеть следующим образом

§A) Лента разбита на ячейки и каждая ячейка может содержать пустой символ 0, либо непустой символ 1, либо символ пробела; §B) Лента разбита на ячейки и каждая ячейка может содержать пустой символ 0, либо непустой символ 1; §C) Лента разбита на равномерные интервалы с ячейками; §D) Устройство памяти разбита на ячейки и каждая ячейка может содержать пустой символ 0, либо непустой символ 1; §E) Лента разбита на ячейки и каждая ячейка может вывести на печать пустой символ 0, либо непустой символ 1;

@76. Как записывается выражение $x^2+3x-\sin 2x+e^x$ в MS Excel?

§A) $x^2+3x-\sin(2x)+\exp(x)$; §B) $x^2+3x-\sin(2*x)+\exp(x)$; §C) $x^2+3*x-\sin(2*x)+\exp(x)$; §D) $x^2+3x-\sin(2*x)+\exp(x)$; §E) $x^2+3*x-\sin(2*x)+e(x)$;

@77. Если для некоторого натурального числа n область определения функции есть подмножество множества $\{0, 1, \dots, n\} \times \{0, 1\}$, а область значений есть подмножество множества $\{0, 1\} \times \{0, 1\}^n$, то такое устройство называется

§A) Машина Черча; §B) Преобразователь рекурсивных функций; §C) Примитивно-рекурсивные машины; §D) Машина Тьюринга; §E) Машина Поста;

@78. В пяти килобайтах

§A) 5000 байт; §B) 5124 байт; §C) 500 байт; §D) 5000 бит; §E) 5120 байт;

@79. Формула Excel может включать

§A) Арифметические операции, переменные; §B) Адреса ячеек, арифметические операции; §C) Арифметические и логические операции, адреса ячеек; §D) Операнды, переменные, адреса ячеек; §E) Символы, операнды, операции;

@80. Для построения блок-схем используются следующие геометрические фигуры:

§A) Эллипс, параллелограмм, прямоугольник, ромб, шестиугольник; §B) Параллелограмм, прямоугольник, ромб, шестиугольник; §C) Эллипс, прямоугольник, ромб, шестиугольник; §D) Эллипс, параллелограмм, прямоугольник, шестиугольник; §E) Эллипс, параллелограмм, прямоугольник, ромб;

@81. Как записывается выражение $x/3+4x^3-\cos x-1.4$ в MS Excel?

§A) $x/3+4*x^3-\cos(x)-1.4$; §B) $x^3+4*x^3-\sin(x)+1.4$; §C) $x\backslash 3+4*3^x-\cos(x)+1.4$; §D) $x^*3+4*x^3-\cos(x)-1.4$; §E) $x\backslash 3+4*x^3-\cos x-1.4$;

@82. Определите результат работы следующей программы:

float x; x=3.14; int y=x+5

§A) y=8.14; §B) y=5.14; §C) y=3.14; §D) y=8; §E) y=5;

@83. Как записывается выражение $\ln x+5(3x+1.5)-|2x|$ в MS Excel?

§A) $\ln(x)+5*(x+1.5)-\text{abs}(2*x)$; §B) $\ln(x)+5*(3*x+1.5)-\text{abs}(2*x)$; §C) $\ln x+5(3*x+1.5)-\text{abs}(2*x)$; §D) $\ln(x)+5(3x+1.5)-\text{abs}(2x)$; §E) $\ln(x)+5(3*x+1.5)-\text{abs}(2*x)$;

@84. Как записывается выражение $x^5 + \arctg(x+\pi)$ на языке BASIC?

- \$A) x^5+atan(x+ PI); \$B) x^5+arctan(x+Pi); \$C) x*5+atn(x+3.14); \$D) pow(x;5)+atan(x+M_PI); \$E)
pow(x*5+arctg(x+pi);

@85. При $x=\pi$ результатом работы оператора *if* $x>3$ then $y=\sin(x)-\cos(x)$ *else* if $x=3$ then $y=(\sin(x))^4-\cos(x)+1$ *else* $y=\exp(x^2-PI)$ будет функция

- \$A) y=sin^4x-cosx+1; \$B) y=sinx-cos^2x; \$C) y=exp(x^2- PI); \$D) y=sinx-cosx; \$E) y=exp(x^2-PI);

@86. Результатом работы программы $a=19; b=5; z=a\%b$ будет

- \$A) z=3; \$B) z=3,8; \$C) z=4; \$D) z=5; \$E) 3,6;

@87. Переведите число 765 из восьмеричной системы счисления в десятичную систему счисления

- \$A) 536; \$B) 497; \$C) 496; \$D) 501; \$E) 761;

@88. Переведите число 1101010 из двоичной системы счисления в десятичную систему счисления

- \$A) 105; \$B) 108; \$C) 216; \$D) 1024; \$E) 106;

@89. Переведите число 10111010 из двоичной системы счисления в восьмеричную систему счисления

- \$A) 172; \$B) 117; \$C) 272; \$D) 2224; \$E) 101011;

@90. Переведите число 7653 из восьмеричной системы счисления в десятичную систему счисления

- \$A) 1506; \$B) 4011; \$C) 4016; \$D) 4010; \$E) 8022;

@91. Переведите число 3473 из восьмеричной системы счисления в десятичную систему счисления

- \$A) 1851; \$B) 1051; \$C) 1151; \$D) 1117; \$E) 1850;

@92. Переведите число 134 из десятичной системы счисления в двоичную систему счисления

- \$A) 110100011; \$B) 10100110; \$C) 1101011; \$D) 10000110; \$E) 10110011;

@93. Переведите число 345 из десятичной системы счисления в восьмеричную систему счисления

- \$A) 226; \$B) 530; \$C) 156; \$D) 100110101; \$E) 531;

@94. Выполнить действия в двоичной системе счисления 1100110+1010101

- \$A) 1010101010; \$B) 101110111; \$C) 100010001; \$D) 100110101; \$E) 100010101;

@95. Выполнить действия в восьмеричной системе счисления 23654+7654321

- \$A) 5647372; \$B) 7700175; \$C) 7610175; \$D) 6710175; \$E) 7677975;

@96. Выполнить действия в двоичной системе счисления 11010x101

- \$A) 10000010; \$B) 10101010; \$C) 10100010; \$D) 10100010; \$E) 100011101;

@97. Выполнить действия в восьмеричной системе счисления 234x76

- \$A) 22711; \$B) 21710; \$C) 22712; \$D) 22710; \$E) 9672;

@98. Что такое система счисления?

- \$A) Последовательность числа 1, 2, 3, 4, 5, 6, 7, 8, 9, 0; \$B) Правила арифметических действий; \$C) Компьютерная программа для арифметических вычислений; \$D) Эта знаковая система, в которой числа записываются по определенным правилам, с помощью знаков некоторого алфавита, называемых цифрами; \$E) Определенная запись числа с помощью некоторых индексов, называемых позиционными системами;

@99. Основанием системы счисления - это

§A) Последовательность числа 1, 2, 3, 4, 5, 6, 7, 8, 9, 0; §B) Арифметическая основа ЭВМ; §C) Количество цифр, используемых для записи чисел; §D) Отношение значений единиц соседних разрядов; §E) Сумма всех цифр позиционной системы счисления;

@J00. Какое значение будет иметь переменная y в выражение $y=2^5+12/(8-2)-2^3$?
§A) -7.25; §B) 4; §C) 27; §D) 6; §E) 8;

@J01. Какое значение будет иметь y в выражение $y=\text{Корень}(17-8)+\text{EXP}(0)-5$?
§A) -1; §B) 1; §C) 4; §D) 3.47; §E) -2;

@J02. Какое значение будет иметь y в выражение $y=\text{SIN}(1)+\text{COS}(1)+\text{ABS}(-7)$?
§A) 3,14; §B) -7; §C) -8; §D) 8; §E) 0;

@J03. Результатом работы программы $x=2; y=x^4+(-1)^{-1}; z=x^4+y^4; cout << "z = " << z;$ будет
§A) $z=3,14$; §B) $z=13$; §C) $z=5$; §D) $z=8$; §E) $z=29$;

@J04. Результатом работы программы $x=4; y=x^4+1; z=\text{sqrt}(3*x+2*y+1); cout << "z = " << z;$ будет
§A) $z=3$; §B) $z=13$; §C) $z=7$; §D) $z=8$; §E) $z=5$;

@J05. Сколько раз повторяется следующий цикл: $\text{for}(k=0; k<=25; k++)$?
§A) 24 раз; §B) 26 раз; §C) 25 раз; §D) 20 раз; §E) 29 раз;

@J06. Результатом работы программы $a=8; b=3; a=a-b; b=a+b; a=b-a; cout << "a = " << a;$ $cout << "b = " << b;$ будет
§A) $a=3; b=8$; §B) $a=8; b=3$; §C) $a=5; b=3$; §D) $a=5; b=8$; §E) $a=0; b=5$;

@J07. Результатом выполнения программы $\text{int } a=15, b=9, max; max=(b>a)?b:a;$ $cout << "max = " << max;$ будет
§A) $\text{max}=a$; §B) $\text{max}=a-15$; §C) $\text{max}=15$; §D) $\text{max}=24$; §E) $\text{max}=6$;

@J08. Следование, ветвление и цикл называют
§A) Базовыми конструкциями языка; §B) Основные алгоритмы языка; §C) Структура программы; §D) Структура алгоритма; §E) Элементы языка;

@J09. Конструкция, представляющая собой последовательное выполнение двух или более операторов - это
§A) Алгоритм; §B) Цикл; §C) Ветвление; §D) Следование; §E) Последовательное;

@J10. Конструкция, которая задает выполнение либо одного, либо другого оператора в зависимости от выполнения какого-либо условия - это
§A) Цикл; §B) Алгоритм; §C) Последовательное; §D) Следование; §E) Ветвление;

@J11. Конструкция, которая задает многократное выполнение операторов - это
§A) Следование; §B) Цикл; §C) Ветвление; §D) Алгоритм; §E) Последовательное;

@J12. Проверка условия выполняется на каждой итерации цикла либо до тела цикла, либо после тела цикла и они называются
§A) Цикл с предусловием, цикл с постусловием; §B) Цикл с предусловием, цикл с после условием; §C) Цикл с начальным условием, цикл с постусловием; §D) Цикл с началом, цикл с концом; §E) Цикл с одним условием, цикл с нескольким условием;

@J13. Способ выписывания простых чисел в интервале от 1 до некоторого N , называется
§A) Схема Горнера; §B) Машина Тьюринга; §C) Решето Эратосфена; §D) Машина Поста; §E) Нет правильного ответа;

@J14. Любой алгоритм существует не сам по себе, а предназначен для
§A) Определенного исполнителя; §B) Решение задачи; §C) Наглядного представления решения задачи; §D) Вычисления функций; §E) Решения круга определенных задач;

@J15. Объекты, над которыми исполнитель может совершать действия, образуют
\$A) Постановку задачи; \$B) Алгоритма решение задачи; \$C) Схема выполнения операций; \$D) Среду исполнителя; \$E) Блок-схему решения задачи;

(@J16. Задачи, для которых невозможно построить процедуру решения, называются
\$A) Некорректные задачи; \$B) Алгоритмически неразрешимые задачи; \$C) Нестандартные задачи; \$D) Задачи, не имеющие решения; \$E) Не вычислимые функции;

(@J17. Слово, к которому применяется алгоритм, называется
\$A) Обработкой информации; \$B) Исходным алгоритмом; \$C) Вводом данных; \$D) Трансляцией информации; \$E) Входным словом;

@J18. Любой алфавит можно заменить другим алфавитом и такая замена является
\$A) Трансляция; \$B) Замена операции; \$C) Кодирование; \$D) Замена оператора; \$E) Декодирование;

@J19. В каждой машине Тьюринга имеются следующие элементы
\$A) Лента, автомат; \$B) Ячейки, память; \$C) Память, оператор, вывод; \$D) Операции, символы, программа; \$E) Процессор, память, ввод, вывод;

@J20. Всякий алгоритм может быть реализован соответствующей машиной Тьюринга. Этот тезис принадлежит
\$A) Клини; \$B) Колмогорову; \$C) Посту; \$D) Тьюрингу; \$E) Черчу;

@J21. Проблема распознавания выводимости алгоритмически неразрешима. Этот тезис принадлежит
\$A) Клини; \$B) Посту; \$C) Колмогорову; \$D) Тьюрингу; \$E) Черчу;

@J22. Все зависимости от выбранной формы записи элементарные шаги алгоритма при укрупнении обединяются в алгоритмические конструкции:
\$A) Последовательные, ветвящиеся, циклические; \$B) Последовательные, ветвящиеся, циклические, рекурсивные; \$C) Ветвящиеся, циклические, рекурсивные; \$D) Последовательные, циклические, рекурсивные; \$E) Последовательные, ветвящиеся, рекурсивные;

@J23. Существует следующие способы для записи одного и того же алгоритма:
\$A) Запись в виде блок-схемы, запись на алгоритмическом языке, представление алгоритма в виде машины Тьюринга или машины Поста; \$B) Последовательные, ветвящиеся, циклические, рекурсивные; \$C) Текстовая форма, запись в виде блок-схемы, запись на алгоритмическом языке, представление алгоритма в виде машины Тьюринга или машины Поста; \$D) Текстовая форма, представление алгоритма в виде машины Тьюринга или машины Поста; \$E) Последовательные, ветвящиеся, рекурсивные, текстовая форма, запись в виде блок-схемы, запись на алгоритмическом языке, представление алгоритма в виде машины Тьюринга или машины Поста;

@J24. В 1969 году Эдсгер Дейкстра доказал, что для записи любого алгоритма достаточны следующие алгоритмические конструкции:

\$A) Последовательные, ветвящиеся, циклические; \$B) Последовательные, ветвящиеся, циклические, рекурсивные; \$C) Текстовая форма, запись в виде блок-схемы, запись на алгоритмическом языке, представление алгоритма в виде машины Поста; \$D) Текстовая форма, представление алгоритма в виде машины Тьюринга; \$E) Последовательные, ветвящиеся, текстовая форма, запись в виде блок-схемы;

@J25. С каждой машиной Тьюринга связаны два конечных алфавита:
\$A) Цифры, буквы; \$B) Алфавит входных символов, алфавит выходных символов; \$C) Алфавит исходных данных, алфавит выходных данных; \$D) Алфавит входных символов, алфавит состояний; \$E) Вход, выход;

@J26. В машине Тьюринга слева и справа от входного слова находятся только

§A) Ячейки с буквами; §B) Символы арифметических операций; §C) Алфавит выходных данных; §D) Ячейки с цифрами; §E) Пустые ячейки;

@127. Чем отличаются состояния в машине Тьюринга от состояний в машине Поста?

§A) В машине Тьюринга состояние задает характер движения головки. В машине Поста состояние описывает местонахождение ячейки и состояние ленты; §B) В машине Тьюринга состояние определяет, что следует записать в обозреваемую ячейку для каждого определенного символа, задает характер движения головки, указывает новое состояние машины. В машине Поста состояние описывает местонахождение каретки и состояние ленты; §C) В машине Тьюринга состояние указывает адрес головки. В машине Поста состояние описывает местонахождение ячейки и состояние ленты; §D) В машине Тьюринга состояние определяет движение головки. В машине Поста состояние описывает местонахождение ячейки; §E) В машине Тьюринга состояние определяет число ячеек и адрес головки. В машине Поста состояние описывает число ячейки и состояние ленты;

Итоговые оценки студентов

Буквенное обозначение итоговых оценок студентов и их цифровые эквиваленты:

Буквенная оценка	Цифра	Общий балл	Традиционная оценка
A	4	$95 \leq A \leq 100$	отлично
A-	3,67	$90 \leq A < 95$	
B+	3,33	$85 \leq B < 90$	хорошо
B	3	$80 \leq B < 85$	
B-	2,67	$75 \leq B < 80$	
C+	2,33	$70 \leq C < 75$	удовлетворительно
C	2	$65 \leq C < 70$	
C-	1,67	$60 \leq C < 65$	
D+	1,33	$55 \leq D < 60$	
D	1	$50 \leq D < 55$	
Fx	0	$45 \leq Fx < 50$	неудовлетворительно
F	0	$0 < F < 45$	

Критерии выведения итоговой оценки промежуточной аттестации:

«Отлично» - средняя оценка $\geq 3,67$.

«Хорошо» - средняя оценка $\geq 2,67$ и $\leq 3,33$.

«Удовлетворительно» - средняя оценка $\geq 1,0$ и $\leq 2,33$.

«Неудовлетворительно» - средняя оценка < 0 .