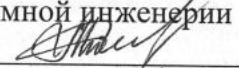


Государственное образовательное учреждение
«ПРИДНЕСТРОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
им. Т.Г. Шевченко»
Рыбницкий филиал
Кафедра «Информатика и программная инженерия»

«УТВЕРЖДАЮ»

Заведующий кафедрой информатики и
программной инженерии

доцент  Л.А. Тягульская

Протокол № 2

« 19 » сентября 2024 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

ПО ДИСЦИПЛИНЕ

«Теория вычислительных процессов»

Направление подготовки:

2.09.03.04 «Программная инженерия»

Профиль подготовки:

«Разработка программно-информационных систем»

Квалификация (степень) выпускника:

Бакалавр

Форма обучения:

очная

Год набора:

2023

Разработал:

доцент


(должность, подпись, ФИО)

Козак Л.Я.

Рыбница, 2024 г.

ПАСПОРТ фонда оценочных средств по учебной дисциплине

1. В результате изучения дисциплины «Теория вычислительных процессов» у обучающихся должны быть сформированы следующие компетенции:

Категория (группа) универсальных компетенций	Код и наименование универсальной компетенции	Код и наименование индикатора достижения универсальной компетенции
Системное и критическое мышление	УК-1. Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	ИД УК-1.1. Знает принципы сбора, отбора и обобщения информации. ИД УК-1.2. Умеет соотносить разнородные явления и систематизировать их в рамках избранных видов профессиональной деятельности. ИД УК-1.3. Имеет практический опыт работы с информационными источниками, опыт научного поиска, создания научных текстов
	ОПК-2. Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности	ОПК-2.1. Знает современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности. ОПК-2.2. Умеет выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности. ОПК-2.3. Имеет навыки применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности.
	ОПК-7. Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой	ОПК-7.1. Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий. ОПК-7.2. Умеет 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. ОПК-7.3. Имеет навыки программирования, отладки и тестирования прототипов программно-технических комплексов задач.

Программа оценивания контролируемой компетенции:

№ п/п	Контролируемые модули, разделы (темы) дисциплины и их наименование	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
1.	Теория схем программ.	УК-1. ОПК-2. ОПК-7.	– текущий – контроль выполнения лабораторных и индивидуальных заданий; посещение занятий; – рубежный – контроль выполнения лабораторных и индивидуальных заданий; контроль выполнения домашнего задания; выполнение контрольной работы; посещение занятий; – итоговый осуществляется посредством тестирования, экзамена
2.	Семантическая теория программ.	УК-1. ОПК-2. ОПК-7.	
3.	Модели вычислительных процессов.	УК-1. ОПК-2. ОПК-7.	
4.	Сети Петри.	УК-1. ОПК-2. ОПК-7.	

«УТВЕРЖДАЮ»
зав. кафедрой ИиИИ,
доцент  Л.А. Тягульская
« 19 » сентября 2024 г.

**Перечень вопросов к экзамену
по дисциплине «Теории вычислительных процессов»
для студентов II курса**

**направления «Программная инженерия»
профиля «Разработка программно-информационных систем»**

1. Понятие схемы программ. Класс стандартных схем программ. Графовая, линейная формы стандартной схемы.
2. Интерпретация стандартных схем. Протокол выполнения программы. Главные свойства стандартных схем.
3. Рекурсивные схемы. Трансляция схем программ. Линейные унарные рекурсивные схемы. Схемы с процедурами.
4. Трансляция рекурсивных схем в схемы с процедурами. Частичная трансляция схем с процедурами.
5. Обогащенные схемы. Класс счетчиковых схем, класс магазинных схем, класс схем с массивами. Трансляция обогащенных схем.
6. Структурированные схемы. Трансляция структурированных схем в стандартные.
7. Основные понятия. Граф существования процесса. Свойства и классификация процессов.
8. Основные определения: граф сети Петри, маркировка, выполнение сети, множество достижимости.
9. Сети Петри для моделирования. Особенности сетей Петри.
10. Анализ сетей Петри. Задачи анализа сетей Петри: безопасность, ограниченность, сохранение, активность, достижимость и покрываемость.
11. Независимые и взаимодействующие вычислительные процессы. Критические интервалы. Взаимное исключение.
12. Средства синхронизации и связи. Блокировка памяти. Семафоры Дейкстры. Мьютексы. Задача «поставщик/потребитель».
13. Задача «читатели/писатели». Мониторы Хоара. Почтовые ящики. Конвейеры и очереди сообщений.
14. Понятие тупиковой ситуации. Повторно используемые и потребляемые ресурсы. Модель Холта. Причины возникновения тупиковых ситуаций.
15. Формальные модели для изучения проблемы тупиковых ситуаций: сети Петри, вычислительные схемы, модели пространства состояний системы.
16. Методы борьбы с тупиками: предотвращение тупиков, обход тупиков, обнаружение тупика.
17. Семантика языка программирования. Методы формального определения семантики.
18. Атрибутивные грамматики. Операционная семантика.
19. Денотационная семантика. Аксиоматическая система Хоара.
20. Верификация программ. Метод индуктивных утверждений. Правила верификации Хоара.

Экзаменатор
доцент  Л.Я. Козак

«УТВЕРЖДАЮ»

зав. кафедрой ИиПИ,

доцент *Л.А. Тягульская*

« 19 » сентября 2024 г.

**Образец тестовых заданий для проведения итогового контроля
по итогам освоения дисциплины, а также для контроля самостоятельной работы студента
по дисциплине «Теория вычислительных процессов»
для студентов II курса
направления «Программная инженерия»
профиля «Разработка программно-информационных систем»**

Указания: Напишите Вашу фамилию, номер группы и дату. Для ответа на вопрос с выбором варианта ответа достаточно написать номер вопроса и рядом букву, обозначающую правильный вариант из предложенных в тесте ответов на вопрос.

1. Трассировочная таблица может использоваться...
 - a) для определения того, что вычисляет программа;
 - b) для определения инварианта цикла;
 - c) для доказательства правильности программы.
2. Принцип простой индукции используется для доказательства высказываний...
 - a) о натуральных числах;
 - b) о целых числах;
 - c) о вещественных числах;
 - d) о неотрицательных числах.
3. При использовании принципа модифицированной простой индукции мы предполагаем, что высказывание истинно для $S(n)$ и на основе этого пытаемся доказать справедливость $S(n+1)$. Это допущение называется индукции.
4. Выберите неправильный ответ. Существуют следующие виды индукции, используемые для доказательства высказываний:
 - a) простая;
 - b) модифицированная простая;
 - c) простая нисходящая;
 - d) строгая нисходящая;
 - e) строгая;
 - f) модифицированная строгая.
5. Выберите правильное утверждение:
 - a) метод индукции является обобщением метода индуктивных утверждений при доказательстве правильности;
 - b) метод индуктивных утверждений является обобщением метода индукции при доказательстве правильности;
 - c) метод индукции и метод индуктивных утверждений являются самостоятельными и независимыми.
6. При использовании метода индуктивных утверждений на блок-схему не выносятся...
 - a) утверждение о правильности;
 - b) утверждение о конечности;
 - c) утверждение-инвариант цикла;
 - d) утверждение о входных данных.
7. Дополните недостающее слово. При доказательстве правильности программ методом индуктивных утверждений утверждения записываются в качестве
8. Выберите правильное утверждение:
 - a) тестирование гарантирует правильность программы;
 - b) доказательство методом индуктивных утверждений гарантирует правильность программы;
 - c) тестирование и доказательство методом индуктивных утверждений гарантирует правильность программы;
 - d) ни тестирование, ни доказательство методом индуктивных утверждений не гарантирует правильность программы.
9. Выберите все правильные ответы

- a) на блок-схеме порядок выполнения явно определяется стрелками;
- b) в программах порядок выполнения явно определяется использованными управляющими структурами языка;
- c) в программах порядок выполнения неявно определяется использованными управляющими структурами языка;
- d) на блок-схеме порядок выполнения неявно определяется стрелками.

10. С помощью сетей Петри нельзя промоделировать задачу...

- a) о взаимном исключении;
- b) о чтении/записи;
- c) о чтении/записи с ограниченным количеством процессов чтения;
- d) о чтении/записи с не ограниченным количеством процессов чтения;
- e) о производителе/потребителе;
- f) о нескольких производителях / нескольких потребителях
- g) о производителе/потребителе с ограниченным буфером

11. Задания на определение зависимости одних явлений от других. Выберите правильный ответ, используя схему

ответ	1	2	вывод
A	+	+	+
B	+	+	-
C	+	-	-
D	-	+	-
E	-	-	-

Метод индуктивных утверждений нельзя использовать для доказательства правильности программ ПОТОМУ ЧТО они, в отличие от блок-схем, могут содержать обращение к подпрограммам.

12. Доказывая конечность программы (при доказательстве правильности программы методом индуктивных утверждений) используем следующие индуктивные утверждения:

- a) утверждение о правильности;
- b) утверждение о конечности;
- c) инвариант цикла;
- d) утверждение о входных данных.

13. О потенциальной опасности тупика свидетельствует следующая характеристика графа Ресурсы-Процессы:

- a) наличие цепей в графе;
- b) наличие простых цепей в графе;
- c) наличие циклических маршрутов в графе;
- d) наличие кратчайшего пути в графе;
- e) наличие в графе пути с минимальным весом.

14. Суть метода глобального предотвращения тупиков заключается в следующем:

- a) процесс запрашивает ресурсы в количестве, существенно превышающем необходимое;
- b) процесс не запускается, пока запрошенные ресурсы не будут иметься в наличии;
- c) процесс приостанавливает все другие процессы, запрашивающие те же ресурсы;
- d) процесс отбирает требуемые ресурсы у других процессов, владеющих ими;
- e) процесс обращается за поиском требуемых ресурсов в Интернет.

15. Суть метода упорядоченных ресурсов состоит в следующем:

- a) процесс может получить ресурс из класса с данным номером, если он освободил ресурс из класса с меньшим номером;
- b) процесс может получить ресурс из класса с данным номером, если он заявил об этом запросе до начала выполнения;
- c) процесс может получить ресурс из класса с данным номером, если другие ресурсы этого класса не захвачены другими процессами;
- d) процесс не может получить ресурс из класса с данным номером, если он уже владеет ресурсами из данного класса;
- e) процесс не может получить ресурс из класса с данным номером, если этот класс не принадлежит к числу заранее зарезервированных классов.

16. При переводе блок-схемы в сеть Петри 1 узел блок-схемы заменяется на...

- a) 1 переход;
- b) 2 перехода;
- c) 1 или 2 перехода;
- d) 1 позицию;
- e) 2 позиции;
- f) 1 или 2 позиции.

17. Расширение модели сети Петри приводит к:

- a) уменьшению мощности разрешения;
- b) увеличению мощности разрешения;
- c) уменьшению мощности моделирования;

- d) увеличению мощности моделирования;
- e) не влияет на мощность разрешения;
- f) ограничению модели сети Петри.

18. Дополните недостающие слова. Граф сети Петри можно записать как $G = (V, A)$, где $V = \{v_1, v_2, \dots, v_s\}$ – вершин, $A = \{a_1, a_2, \dots, a_r\}$ – дуг.

19. В сети Петри возможны события

- a) мгновенные;
- b) примитивные;
- c) длительные;

- d) неодновременные;
- e) непримитивные.

20. Выберите наиболее правильный ответ. Чаще всего сети Петри используются для моделирования

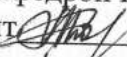
- a) аппаратного обеспечения;
- b) программного обеспечения;
- c) механизмов синхронизации;
- d) процессов конвейерной обработки;
- e) химических реакций;
- f) транспортных потоков.

Ключи к тестовым заданиям

№ задания	ключи
1	a, c
2	d
3	гипотезой
4	f
5	b
6	b
7	точек
8	d
9	a, c
10	b
11	c
12	d
13	a
14	b
15	a
16	d
17	a
18	множество
19	e
20	b

«УТВЕРЖДАЮ»

зав. кафедрой ИиПИ,

доцент  Л.А. Тягульская

« 19 » сентября 2024 г.

**Темы для докладов
по дисциплине «Теория вычислительных процессов»
для студентов II курса направления «Программная инженерия»**

Тема 1. Математическая индукция. Введение: основные направления исследований, связанные с доказательством правильности программ (методы доказательства, конструирование программ и языков, механизация процесса доказательства правильности. Простая индукция: принцип простой индукции, принцип модифицированной простой индукции, доказательство высказываний, относящихся к программам для вычислительных машин. Строгая версия математической индукции: принцип строгой индукции. Обобщенная индукция: принцип обобщенной индукции.

Тема 2. Доказательство правильности блок-схем программ. Основные принципы доказательства правильности для блок-схем. Понятие утверждения о правильности программ, инварианта цикла, утверждения о входных данных и утверждения о конечности программы. Дополнительные примеры доказательства правильности блок-схем программ.

Тема 3. Метод индуктивных утверждений – основные определения и теоремы, описание метода. Частичная и полная правильность программы. Доказательство частичной правильности и конечности. Сокращенные доказательства правильности..

Тема 4. Доказательство правильности программ, написанных на обычных языках программирования. Примеры доказательства правильности программ, написанных на Фортране. Примеры доказательства правильности программ, написанных на ПЛ/1. Особенности доказательства правильности программ, написанных на языках программирования Visual Basic и Си.

Тема 5. Формализация доказательства с помощью индуктивных утверждений. Аксиоматический подход к доказательству частичной правильности. Доказательство правильности как часть процесса программирования.

Тема 6. Доказательство правильности рекурсивных программ. Упрощенный язык программирования для иллюстрации понятия рекурсии. Структурная индукция. Более трудные примеры доказательства правильности программ методом структурной индукции. Структурная индукция для нерекурсивных программ.

Тема 7. Современные исследования, связанные с доказательством правильности программ. Методы доказательства. Конструирование программ и языков. Механизация процесса доказательства правильности.

Тема 8. Сети Петри как инструмент моделирования систем. Введение. Литература. Понятие моделирования. Природа систем. Применение теории сетей Петри. Прикладная и чистая теории сетей Петри.

Тема 9. Основные определения и понятия. Структура сетей Петри. Графы сетей Петри. Маркировка сетей Петри. Правила выполнения сетей Петри. Пространство состояний сетей Петри. Альтернативные формы определения сетей Петри.

Тема 10. Сети Петри для моделирования. Области применения сетей Петри. Принципы построения сетей Петри: события и условия, одновременность и конфликт аппаратного обеспечения ЭВМ: конечные автоматы, ЭВМ с конвейерной обработкой, кратные функциональные блоки. Моделирование сетями Петри программного обеспечения ЭВМ: блок-схемы, обеспечение параллелизма, задачи синхронизации. Другие системы с асинхронными параллельными взаимодействующими процессами.

Тема 11. Задачи и методы анализа сетей Петри. Задачи анализа сетей Петри: безопасность, ограниченность, сохранение, активность, достижимость и покрываемость, эквивалентность и подмножество. Методы анализа сетей Петри. дерево достижимости, его построение и ограниченность применения, матричные уравнения.

Тема 12. Расширенные и ограниченные модели сетей Петри. Границы возможностей моделирования с помощью сетей Петри. Взаимосвязь мощности моделирования и мощности разрешения. Виды расширений. Подклассы сетей Петри.

Тема 13. Модели параллельных вычислений. Понятия эквивалентности и включения моделей. Соотношения между классами моделей и место сетей Петри в полной иерархии моделей.

«УТВЕРЖДАЮ»
зав. кафедрой ИИПИ,
доцент Л.А. Тягульская
« 19 » сентября 2024 г.

**Образец контрольной работы по разделу «Сети Петри»
по дисциплине «Теория вычислительных процессов»
для студентов II курса
направления «Программная инженерия»
профиля «Разработка программно-информационных систем»**

Образец контрольной работы по разделу «Сети Петри»

Для предлагаемой сети Петри, заданной в виде двудольного ориентированного мультиграфа, выполнить следующее:

- 1) определить сеть Петри в виде $C = (P, T, I, O)$;
 - 2) определить расширенные входную и выходную функцию;
 - 3) определить мультиграф в виде $G = (V, A)$;
 - 4) нарисовать граф инверсной сети Петри и описать ее как $C = (P, T, I, O)$;
 - 5) нарисовать граф двойственной сети Петри и описать ее как $C = (P, T, I, O)$;
 - 6) выполнить сеть Петри, записав последовательность переходов $\sigma = t_{j1}, \dots, t_{jk}$ и последовательность маркировок $\mu^0, \dots, \mu^k$;
 - 7) построить дерево достижимости и определить тип всех его вершин;
 - 8) по дереву достижимости проверить свойства сети Петри: является ли она
а) безопасной, б) ограниченной, в) активной, г) строго сохраняющей;
 - 9) для матричного представления сети Петри определить матрицы D^- , D^+ , D и вектор запусков последовательности $f(\sigma)$ для σ из п.б.
- Вычислить $\mu' = \mu^0 + f(\sigma) \cdot D$.

