

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Рукович Александр Владимирович

Должность: Директор

Дата подписания: 30.05.2025 14:59:52

Уникальный идентификатор:

f45eb7c44954саас05ea7d4f32eb8d7d6b3cb963e6d9b4bda094afdda9fb705f

Министерство науки и высшего образования Российской Федерации  
Технический институт (филиал) федерального государственного автономного  
образовательного учреждения высшего образования «Северо-Восточный федеральный  
университет имени М.К. Аммосова» в г. Нерюнгри

## КАФЕДРА МАТЕМАТИКИ И ИНФОРМАТИКИ

### ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

#### Б1.В.ДВ.01.02 Системное программирование

для программы бакалавриата

по направлению подготовки 09.03.03 «Прикладная информатика»

Направленность программы: Прикладная информатика в менеджменте

Форма обучения: заочная

Нерюнгри 2020

УТВЕРЖДЕНО  
на заседании выпускающей кафедры

«22» апреля 2020 г., протокол № 10

Заведующий кафедрой МиИ  В.М. Самохина  
«22» апреля 2020 г.

СОГЛАСОВАНО:

Эксперты<sup>1</sup>:

Самохина В.М., зав. кафедрой МиИ, ТИ (ф) СВФУ 

Ф.И.О., должность, организация, подпись

Юданова В.В., старший преподаватель кафедры МиИ, ТИ (ф) СВФУ 

Ф.И.О., должность, организация, подпись

СОСТАВИТЕЛЬ:

Похорукова М.Ю., доцент кафедры МиИ, ТИ (ф) СВФУ 

Ф.И.О., должность, организация, подпись



<sup>1</sup> Эксперт первый: со стороны выпускающей кафедры (или работодатель). Эксперт второй: со стороны обеспечивающей кафедры.

## Паспорт фонда оценочных средств

### Б1.В.ДВ.01.02 Системное программирование

| Коды оцениваемых компетенций                                                                                                                                                                                                                | Показатель оценивания (по п.1.2.РПД)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | Уровни освоения | Критерии оценивания (дескрипторы)                                                                                                                                                                                                                                                                                                                                                                                                      | Оценка     |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>ПК-2:</b><br>Способен осуществлять проектирование программного обеспечения.<br><br><b>ПК-3:</b><br>Способен осуществлять алгоритмизацию поставленных задач и применять выбранные языки программирования для написания программного кода. | <b>Знать:</b> принципы разработки системного программного обеспечения; особенности выполнения программ, механизмы синхронизации: семафоры, мониторы, сообщения, барьеры.<br><b>Уметь:</b> разрабатывать многозадачное и многопоточное программное обеспечение; использовать аппаратные и программные средства современных операционных систем для управления памятью, файлами, процессами и потоками.<br><b>Владеть:</b> навыками разработки и применения алгоритмических и программных решений в области системного и прикладного программного обеспечения; навыками анализа и проверки программного кода, его отладки на уровне программных модулей и межмодульных взаимодействий. | Освоен          | Обучаемый демонстрирует способность к полной самостоятельности (допускаются консультации с преподавателем по сопутствующим вопросам) в выборе способа решения нестандартных заданий с использованием инструментария современных ИТ. Присутствие сформированной компетенции на высоком уровне, способность к ее дальнейшему саморазвитию и высокой адаптивности практического применения в условиях своей профессиональной деятельности | Зачтено    |
|                                                                                                                                                                                                                                             |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | Не освоено      | Неспособность обучаемого самостоятельно продемонстрировать наличие знаний при решении заданий, которые были представлены преподавателем вместе с образцом их решения. Отсутствие самостоятельности в применении умения к использованию инструментария ИТ для решения задач в профессиональной деятельности и неспособность самостоятельно проявить навык повторения решения поставленной задачи по стандартному образцу.               | Не зачтено |

Федеральное государственное автономное образовательное учреждение  
высшего образования  
«СЕВЕРО-ВОСТОЧНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ ИМЕНИ М.К. АММОСОВА»  
Технический институт (филиал) в г. Нерюнгри

Кафедра математики и информатики

**Лабораторная работа**

В период освоения дисциплины студенты посещают лекционные занятия, самостоятельно изучают дополнительный теоретический материал к лабораторным занятиям. Критериями оценки работы на лабораторных занятиях является: полнота и правильность выполненного задания; степень осознанности, понимания изученного; оформление задания.

Темы лабораторных работ

**Тема 1.** Введение в системное программирование.

**Тема 2.** Особенности выполнения программ. Процессы и потоки.

**Тема 3.** Файловые системы.

**Критерии оценки:**

0 баллов - ставится, если студент не выполнил лабораторную работу.

1 балл - ставится, если студент обнаруживает знание и понимание основных положений лабораторной работы, но при выполнении заданий допущены ошибки или задание выполнено на 50%; оформление работы выполнено недостаточно последовательно (отсутствуют цель/листинг/результаты/выводы).

2 балла - ставится, если студентом при выполнении заданий допущены неточности или задание выполнено на 70%; оформление работы выполнено с ошибками (отсутствуют цель/выводы).

3 балла - ставится, если студент полностью выполнил задание, правильно ответил на теоретические вопросы преподавателя, оформление работы выполнено последовательно и полно (присутствуют цели работы, задания, листинг программ, результаты и выводы).

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«СЕВЕРО-ВОСТОЧНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ ИМЕНИ М.К. АММОСОВА»  
Технический институт (филиал) в г. Нерюнгри

Кафедра математики и информатики

**Самостоятельная работа студента**

Включает проработку конспектов лекций, обязательной и дополнительной учебной литературы в соответствии с планом занятия; выполнение заданий. Основной формой проверки СРС является устный фронтальный опрос на занятии и письменные ответы на вопросы для проверки знаний по теме.

Темы заданий для самостоятельной работы студентов

**Тема 1.** Введение в системное программирование.

**Тема 2.** Особенности выполнения программ. Процессы и потоки.

**Тема 3.** Файловые системы.

Критерии оценки:

0 баллов – самостоятельная работа не выполнена.

1 балл – демонстрирует, лишь поверхностный уровень выполнения работы, в содержании выполнения задания допущены ошибки.

2 балла – ставится тогда, когда студент выполнил самостоятельную работу, но дает не точные ответы на заданные вопросы.

3 балла – ставится тогда, когда студент выполнил самостоятельную работу, показан высокий уровень освоения студентом учебного материала, содержание выполнения задания не содержит ошибок.

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«СЕВЕРО-ВОСТОЧНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ ИМЕНИ М.К. АММОСОВА»  
Технический институт (филиал) в г. Нерюнгри

Кафедра математики и информатики

**Контрольная работа**

Контрольная работа предполагает выполнение письменной работы с обязательными практическими примерами по одной из тем.

Тематика контрольных работ

1. Объекты ядра, процесс выполнения программ.
2. Синхронизация потоков, решение проблем синхронизации.
3. Принципы аппаратуры ввода-вывода, программное обеспечение.
4. Интерфейс операционной системы: основные принципы и стандарты.
5. Межпроцессные взаимодействия: механизмы, каналы, очереди сообщений, разделяемые сегменты памяти, сокеты, вызов удаленных процедур.
6. Драйвера: задачи, классификация и особенности их функционирования, основные свойства и характеристики.
7. Службы, особенности их создания и работы.
8. Подсистема безопасности: цели, защита объектов, аудит, права и привилегии.
9. Процессы и потоки.

Критерии оценки:

| №   | Критерий                                                                                                                                                           |           |
|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| 1.  | Соответствие содержания заявленной теме                                                                                                                            | 16        |
| 2.  | Логичность и последовательность в изложении материала                                                                                                              | 16        |
| 3.  | Способность к работе с литературными источниками, Интернет-ресурсами, справочной и энциклопедической литературой                                                   | 16        |
| 4.  | Способность к выполнению практических заданий по заданной тематике                                                                                                 | 16        |
| 5.  | Использование компьютерных программ при выполнении задания                                                                                                         | 16        |
| 6.  | Анализ полученных результатов, обоснованность выводов                                                                                                              | 16        |
| 7.  | Правильность оформления (наличие всех структурных частей, структурная упорядоченность, ссылки на литературу, цитаты, таблицы, рисунки и т.д.);                     | 16        |
| 8.  | Соответствие оформления правилам компьютерного набора текста (соблюдение объема, шрифтов, интервалов, выравнивания текста на страницах, нумерация страниц и т.д.); | 16        |
| 9.  | Наличие презентационного материала                                                                                                                                 | 16        |
| 10. | Правильность ответов на заданные вопросы по заявленной теме                                                                                                        | 16        |
|     | <b>Итого</b>                                                                                                                                                       | <b>10</b> |

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«СЕВЕРО-ВОСТОЧНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ ИМЕНИ М.К. АММОСОВА»  
Технический институт (филиал) в г. Нерюнгри

Кафедра математики и информатики

**Тестирование**

1. Выберите из списка группы, которые не соответствуют способам написания программ.
  - a) формирование потоков параллельного управления, без учета особенностей архитектур параллельных вычислительных систем или операционных систем
  - b) последовательное программирование с дальнейшим ручным распараллеливанием
  - c) описание параллелизма с использованием явного управления обеспечивается заданием только информационных связей
  - d) последовательное программирование с дальнейшим автоматическим распараллеливанием
  - e) формирование потоков параллельного управления, с учетом особенностей архитектур параллельных вычислительных систем или операционных систем
  - f) описание параллелизма без использования явного управления обеспечивается заданием только информационных связей
2. Выберите определение, которое соответствует параллельному вычислению:
  - a) техника, которая использует преимущества многоядерных или многопроцессорных компьютеров и является подмножеством более широкого понятия многопоточности
  - b) способ организации компьютерных вычислений, при котором программы разрабатываются, как набор взаимодействующих вычислительных процессов, работающих асинхронно и при этом одновременно
  - c) способ организации компьютерных вычислений, при котором программы разрабатываются, как набор самостоятельных вычислительных процессов, работающих последовательно
  - d) техника, которая позволяет максимально оптимизировать использование одноядерных или однопроцессорных
3. Выберите определение, которое соответствует параллельному программированию.
  - a) техника, которая использует преимущества многоядерных или многопроцессорных компьютеров и является подмножеством более широкого понятия многопоточности
  - b) способ организации компьютерных вычислений, при котором программы разрабатываются, как набор самостоятельных вычислительных процессов, работающих последовательно
  - c) техника, которая позволяет максимально оптимизировать использование одноядерных или однопроцессорных
  - d) способ организации компьютерных вычислений, при котором программы разрабатываются, как набор взаимодействующих вычислительных процессов, работающих асинхронно и при этом одновременно
4. М.Флинн выделил следующие классы архитектур:
  - a) MIDS
  - b) MIMD
  - c) SISD
  - d) MIMS
  - e) SIMD
  - f) MISD
  - g) SIDS
5. Какая из архитектур имеет одиночный поток команд и одиночный поток данных?
  - a) MIMS

- b) MIDS
  - c) MIMD
  - d) SISD
6. Какая из архитектур имеет одиночный поток команд и множественный поток данных
- a) MIMS
  - b) MIDS
  - c) SIMD
  - d) SISD
7. Закон Амдала соответствует следующей формулировке...
- a) описывает максимальный теоретический выигрыш в производительности параллельного решения по отношению к лучшему последовательному решению
  - b) описывает фактический выигрыш в производительности параллельного решения по отношению к лучшему последовательному решению
  - c) оценивает минимально допустимое ускорение выполнения параллельной программы, в зависимости от количества одновременно выполняемых потоков вычислений и доли последовательных расчётов
  - d) оценивает максимально допустимое ускорение выполнения параллельной программы, в зависимости от количества одновременно выполняемых потоков вычислений и доли последовательных расчётов
8. Закон Густафсона-Барсиса...
- a) описывает фактический выигрыш в производительности параллельного решения по отношению к лучшему последовательному решению
  - b) оценивает минимально допустимое ускорение выполнения параллельной программы, в зависимости от количества одновременно выполняемых потоков вычислений и доли последовательных расчётов
  - c) описывает максимальный теоретический выигрыш в производительности параллельного решения по отношению к лучшему последовательному решению
  - d) оценивает максимально допустимое ускорение выполнения параллельной программы, в зависимости от количества одновременно выполняемых потоков вычислений и доли последовательных расчётов
9. Выберите из списка компоненты библиотеки PFX.
- a) библиотека CLR
  - b) библиотека TPL
  - c) класс PLINQ
  - d) пул потоков CLR
  - e) пул потоков TPL
  - f) библиотека PLINQ
  - g) библиотека Parallel
  - h) параллельные коллекции
  - i) спин-примитивы
10. Библиотека использующая декларативный подход...
- a) библиотека PLINQ
  - b) библиотека CLR
  - c) библиотека TPL
  - d) библиотека Parallel
11. Какая библиотека позволяет использовать императивный подход?
- a) библиотека PLINQ
  - b) библиотека Parallel
  - c) библиотека TPL
  - d) библиотека CLR
12. С помощью, каких директив можно использовать библиотеку TPL?
- a) using System.Tasks

- b) `using System.Threading.Tasks`
  - c) `using System. Parallel.Task`
  - d) `using System. Tasks.Threading`
  - e) `using System.Threading`
13. Выберите определение соответствующее потоку.
- a) запущенная программа
  - b) управляемая единица исполняемого кода
  - c) специализированная форма многозадачности
  - d) запущенная в фоновом режиме программа
14. Какое из определений соответствует определению многопоточности?
- a) специализированная форма многозадачности
  - b) запущенная в фоновом режиме программа
  - c) запущенная программа
  - d) управляемая единица исполняемого кода
15. Какое из определений соответствует определению процесса?
- a) запущенный в фоновом режиме поток
  - b) специализированная форма многозадачности
  - c) запущенная программа
  - d) управляемая единица исполняемого кода
16. В классе `Thread` содержатся следующие свойства...
- a) `CurrentContext`, `Context`, `IsAlive`, `IsBackground`, `Priority`, `Name`, `CurrentName`, `ThreadState`, `CurrentThread`
  - b) `CurrentContext`, `IsAlive`, `IsBackground`, `Priority`, `Name`, `ThreadState`, `CurrentThread`
  - c) `Context`, `IsAlive`, `IsBackground`, `Priority`, `CurrentName`, `ThreadState`, `CurrentThread`
  - d) `Context`, `IsAlive`, `IsBackground`, `Priority`, `Name`, `ThreadState`, `CurrentThread`
17. Выберите правильную последовательность действий для создания вторичного потока
- a) создается объект `Thread`, Устанавливаются начальные характеристики потока. Создается новый делегат `ParametrizedThreadStart`. Создается метод, который будет точкой входа для нового потока. Вызывается метод `Thread.Start()`
  - b) устанавливаются начальные характеристики потока. Вызывается метод `Thread.Start()`. Создается объект `Thread`, Создается метод, который будет точкой входа для нового потока
  - c) устанавливаются начальные характеристики потока. Создается новый делегат `ParametrizedThreadStart`. Вызывается метод `Thread.Start()`. Создается объект `Thread`, Создается метод, который будет точкой входа для нового потока
  - d) создается метод, который будет точкой входа для нового потока. Создается новый делегат `ParametrizedThreadStart`. Создается объект `Thread`. Устанавливаются начальные характеристики потока. Вызывается метод `Thread.Start()`
18. Какой синтаксис установки приоритета потока, является верным?
- a) `Thread.Priority = Priority.Hight;`
  - b) `Process.GetCurrentProcess().PriorityClass = ProcessPriorityClass.High;`
  - c) `Process.PriorityClass = ProcessPriorityClass.High;`
  - d) `Thread.Priority = ThreadPriority.BelowNormal;`
19. Что из списка соответствует последовательности "Запустить поток, Приостановить выполнение потока, Остановить выполнение потока" и является синтаксически верным?
- a)  
`Thread.Start();`  
`Thread.Pause();`  
`Thread.Abort();`
  - b)  
`Thread.Start();`  
`Thread.Sleep(CurrentThread.Sleep);`  
`Thread.Abort();`

c)  
Thread.Start();  
Thread.Sleep();  
Thread.Cancel()

d)  
Thread.Start();  
Thread.Sleep(1000);  
Thread.Abort()

20. Выберите из списка метод позволяющий запросить поток из пула.

- a) QueueWorkItem()
- b) CurrentQueueUserWorkItem()
- c) CurrentQueueWorkItem()
- d) QueueUserWorkItem()

21. В классе Thread содержатся следующие свойства...

- a) CurrentContext, IsAlive, IsBackground, Sleep, Abort, Join, Suspend
- b) Context, IsAlive, IsBackground, Sleep, Abort, Join, Procced
- c) CurrentContext, IsAlive, IsBackground, Stop, Abort, Join, Suspend
- d) Context, IsAlive, IsBackground, Sleep, Stop, Join, Suspend

22. Выберите верное назначение оператора lock:

- a) предназначен для того, что бы одному потоку не дать войти в важный раздел кода в тот момент, когда в нем находится другой поток
- b) предназначен для использования атомарных операций для переменных, общедоступных нескольким потокам
- c) предназначен для установки приоритета между различными потоками
- d) предназначен для того, что бы контролировать доступ к объектам, предоставляя блокировку объекта одному потоку

23. Выберите верное назначение класса Interlocked.

- a) предназначен для того, что бы контролировать доступ к объектам, предоставляя блокировку объекта одному потоку
- b) предназначен для использования атомарных операций для переменных, общедоступных нескольким потокам
- c) предназначен для того, что бы множество потоков встречались в определенном месте во времени
- d) предназначен для управления доступом к пулу ресурсов

24. Класс Monitor...

- a) предназначен для того, что бы контролировать доступ к объектам, предоставляя блокировку объекта одному потоку
- b) предназначен для предоставления доступа к общему ресурсу только одному потоку синхронизации
- c) предназначен для установки приоритета между различными потоками
- d) предназначен для того, что бы множество потоков встречались в определенном месте во времени

25. Какой из методов возобновляет выполнение потока, стоящего первым в очереди потоков?

- a) PulseAll()
- b) WaitAll()
- c) Pulse()
- d) Wait()

26. Какой из методов переводит поток в режим ожидания и снимает блокировку с объекта, позволяя другому потоку использовать этот объект?

- e) PulseAll()
- f) WaitAll()
- g) Pulse()

h) Wait()

27. Класс Mutex...

- a) предназначен для управления доступом к пулу ресурсов
- b) предназначен для того, что бы одному потоку не дать войти в важный раздел кода в тот момент, когда в нем находится другой поток
- c) предназначен для предоставления доступа к общему ресурсу только одному потоку синхронизации
- d) предназначен для установки приоритета между различными потоками

28. Для чего используется класс Semaphore?

- a) используется для управления доступом к пулу ресурсов
- b) используется для того, что бы одному потоку не дать войти в важный раздел кода в тот момент, когда в нем находится другой поток
- c) используется для того, что бы контролировать доступ к объектам, предоставляя блокировку объекта одному потоку
- d) используется для использования атомарных операций для переменных, общедоступных нескольким потокам

29. Какое из событий позволяет потокам взаимодействовать друг с другом путем передачи сигналов и не использует метод Reset() для перехода в исходное состояние?

- a) ManualResetEvent
- b) ManualResetEventSlim
- c) AutoResetEvent

30. Выберите способы создания объектов класса Task.

- a) с использованием именного метода Action и делегата
- b) с использованием именного делегата
- c) с использованием анонимного делегата
- d) с использованием именного метода
- e) с использованием лямбда-выражения и делегата
- f) с использованием лямбда-выражения и именного метода
- g) с использованием лямбда-выражения и анонимного метода
- h) с использованием метода Action и анонимного метода
- i) с использованием делегата Action и именного метода

1. Какие классы не содержатся в пространстве имен System.Threading.Task?
  - a. TaskFactory
  - b. Task <TResult>
  - c. TaskFactory <TResult>
  - d. TaskScheduler <TResult>
  - e. TaskScheduler
  - b) Task
2. В пространстве имен System.Threading.Task содержатся следующие классы...
  - a) Task <TResult>
  - b) Task
  - c) TaskFactory <TResult>
  - d) TaskScheduler <TResult>
  - e) TaskFactory
  - f) TaskScheduler
3. Создать и запустить задачи возможно с помощью следующих фрагментов кода...
  - a) Task task = new Task (() => Console.WriteLine ("Hello World!"));
  - b) Task task = new Task (Console.WriteLine ("Hello World!")); task.Start();
  - c) Task.Factory.StartNew (() => Console.WriteLine ("Hello World!"));
  - d) Task task = new Task (() => Console.WriteLine ("Hello World!")); task.Start();
  - e) Task.Factory.StartNew (() => Console.WriteLine ("Hello World!")).Start();
  - f) Task.Factory.StartNew (Console.WriteLine ("Hello World!"));
4. Выбери методы и свойства, которые используются для ожидания выполнения задач.
  - a) свойство Result
  - b) метод Stop
  - c) свойство IsCompleted
  - d) метод WaitAll
  - e) метод Wait
  - f) метод Dispose
  - g) метод ContinueWith
5. Какому объекту будет передано необработанное исключение при вызове кода в задаче?
  - a) ArithmeticException
  - b) ApplicationException
  - c) NullReferenceException
  - d) AggregationException
  - e) DivideByZeroException
6. Какую структуру следует использовать для отмены выполнения задач?
  - a) CancellationToken
  - b) CancellThreadToken
  - c) CancellationTaskToken
  - d) CancellToken
  - e) CancellationThreadToken
  - f) CancellTaskToken
7. Выберите правильную форму метода, предназначенного для выполнения продолжения задачи.
  - a) public Task ContinueWith()
  - b) public Task ContinueWith(Action<Task>)
  - c) public Task Continue (Action<Task>)
  - d) public Task Continue ()
8. С помощью какого свойства можно определить, было ли сгенерировано исключение родительской задачей?
  - a) ThreadException
  - b) TaskMainException

- c) Exception
- d) TaskException
- 9. Какая из директив позволяет использовать библиотек Parallel?
  - a) using System.Threading.Parallel
  - b) using System.Threading.Tasks.Parallel
  - c) using System.Threading.Task.Parallel
  - d) using System.Threading.Parallel.Task
- 10. Какие методы и их формы объявления входят в класс Parallel?
  - a) For(Int32, Int32, Action<Int32,ParallelLoopState>)
  - b) Invoke(Action[], ParallelOptions)
  - c) Invoke(Action[])
  - d) Invoke(ParallelOptions, Action[])
  - e) Invoke()
  - f) For(Action, Int32, Int32)
  - g) For(Int32, Int32, Action)
  - h) ForEach(IEnumerable, Action<Int32,ParallelLoopState>)
  - i) ForEach(Action, IEnumerable)
  - j) ForEach(IEnumerable, Action)
- 11. Какой из методов (и его верная форма) из списка выполняет цикл for, обеспечивая возможность параллельного выполнения итераций, а также контроля состояния цикла и управления этим состоянием?
  - a) For(Int32, Int32, Action)
  - b) For(Int32, Int32, Action<Int32,ParallelLoopState>)
  - c) For(Action, Int32, Int32)
  - d) For(Action<Int32,ParallelLoopState>),Int32, Int32)
- 12. Выберите верные формы объявления параллельного цикла foreach.
  - a) ForEach(IEnumerable, Action< ParallelLoopState>)
  - b) ForEach(IEnumerable, Action)
  - c) ForEach(Action, IEnumerable)
  - d) ForEach(IEnumerable, Action<Int32,ParallelLoopState>)
- 13. Выберите верные формы объявления метода Invoke.
  - a) Invoke(Action[], ParallelOptions)
  - b) Invoke(ParallelOptions)
  - c) Invoke(Action[])
  - d) Invoke(ParallelOptions, Action[])
- 14. Метод, который выполняет операцию foreach (For Each в Visual Basic) для объекта IEnumerable, обеспечивая возможность параллельного выполнения итераций...
  - a) ForEach(IEnumerable, Action)
  - b) ForEach(IEnumerable, Action<Int32,ParallelLoopState>)
  - c) ForEach(IEnumerable, Action< ParallelLoopState>)
  - d) ForEach(Action, IEnumerable)

| Процент выполненных тестовых заданий | Количество набранных баллов |
|--------------------------------------|-----------------------------|
| 91% - 100%                           | 10                          |
| 81% - 90%                            | 9                           |
| 71% - 80%                            | 8                           |
| 61% - 70%                            | 7                           |
| 51% - 60%                            | 6                           |
| <50%                                 | 0                           |