

**Федеральное государственное автономное образовательное учреждение  
высшего образования  
"Национальный исследовательский университет  
"Высшая школа экономики"**

Факультет компьютерных наук  
Департамент программной инженерии

**Рабочая программа дисциплины  
Распределённые вычисления**

для образовательной программы «Программная инженерия»  
направления подготовки 09.03.04 «Программная инженерия»  
уровень - бакалавр

Разработчики программы  
Витушкин П.И., [pvitushkin@hse.ru](mailto:pvitushkin@hse.ru)

Одобрена на заседании департамента программной инженерии «\_\_» \_\_\_\_\_ 2018 г.  
Руководитель департамента Авдошин С.М. \_\_\_\_\_

Утверждена Академическим советом образовательной программы  
«\_\_» \_\_\_\_\_ 2018 г., № протокола \_\_\_\_\_

Академический руководитель образовательной программы  
Шилов В.В. \_\_\_\_\_

Москва, 2018

*Настоящая программа не может быть использована другими подразделениями университета  
и другими вузами без разрешения подразделения-разработчика программы.*



## 1 Область применения и нормативные ссылки

Настоящая программа учебной дисциплины "Распределённые вычисления" устанавливает минимальные требования к знаниям и умениям студента и определяет содержание и виды учебных занятий и отчетности.

Программа предназначена для преподавателей, ведущих данную дисциплину, учебных ассистентов и студентов образовательной программы «Программная инженерия» направления подготовки 09.03.04 "Программная инженерия", изучающих дисциплину "Распределённые вычисления".

Программа разработана в соответствии с:

- образовательным стандартом Федерального государственного автономного образовательного учреждения высшего образования «Национальный исследовательский университет «Высшая школа экономики» по направлению 09.03.04 «Программная инженерия»;
- образовательной программой «Программная инженерия» направления подготовки 09.03.04 "Программная инженерия";
- рабочим учебным планом университета по направлению 09.03.04 "Программная инженерия", утверждённым в 2018 г.

## 2 Цели освоения дисциплины

Цели освоения дисциплины "Распределённые вычисления":

- формирование у студентов профессиональных компетенций, связанных с использованием теоретических знаний в области распределённых вычислений (распределённой обработки информации);
- получение практических навыков в области выбора систем распределённой обработки, наилучшим образом реализующих поставленные цели обработки информации с учётом заданных требований;
- развитие умений, основанных на полученных теоретических знаниях, позволяющих на творческом и репродуктивном уровне применять известные системы распределённой обработки информации и в их рамках создавать собственные подсистемы для эффективного решения поставленных задач;
- получение студентам навыков самостоятельной исследовательской работы, предполагающей изучение специфических методов распределённых вычислений, анализа их архитектур и функционирующего на них программного обеспечения, инструментов и средств, необходимых для решения актуальной, в аспекте программной инженерии, задачи выбора рациональных алгоритмов в зависимости от особенностей применения разрабатываемых программ.



### 3 Компетенции обучающегося, формируемые в результате освоения дисциплины

В результате освоения дисциплины студент должен:

- Знать:
  - методы обработки информации в распределённых вычислительных системах;
  - методы хранения информации в распределённых вычислительных системах;
  - методы обмена информацией между компонентами распределённых вычислительных систем.
- Уметь:
  - оценивать системы распределённых вычислений с точки зрения комплексных критериев качества;
  - планировать эксперимент, проводить экспериментальное исследование с помощью имитационных моделей распределённых вычислительных систем.
- Иметь навыки (приобрести опыт) и владеть:
  - методами анализа потоков данных в системах распределённых вычислений различной архитектуры;
  - инструментами замера времени при обработке данных в распределённых системах обработки информации в программных реализациях алгоритмов;
  - основами технологий разработки программ для использования в системах распределённых вычислений.

В результате изучения дисциплины студент осваивает следующие компетенции:

<i>Компетенция</i>	<i>Код по ФГОС/НИУ</i>
Способен учиться, приобретать новые знания, умения, в том числе в области, отличной от профессиональной	УК-1
Способен работать в команде	УК-7
Способен грамотно строить коммуникацию, исходя из целей и ситуации общения	УК-8
Способен критически оценивать и переосмысливать накопленный опыт (собственный и чужой), рефлексировать профессиональную и социальную деятельность	УК-9
Способен осуществлять производственную или прикладную деятельность в международной среде	УК-10
Способен применять основные концепции, принципы, теории и факты, связанные с информатикой при решении научно-исследовательских	ПК-1
Способность к формализации в своей предметной области с учетом ограничений используемых методов исследования	ПК-2
Способен использовать методы и инструментальные средства исследования объектов профессиональной деятельности	ПК-3
Способен обосновать принимаемые проектные решения, осуществлять постановку и выполнение экспериментов по проверке их корректности и эффективности	ПК-4
Способен готовить презентации, оформлять научно-технические отчеты по результатам выполненной работы, публиковать результаты	ПК-5
Способен формализовать предметную область программного проекта и разработать спецификации для компонентов программного продукта	ПК-6



<i>Компетенция</i>	<i>Код по ФГОС/НИУ</i>
Способен выполнить начальную оценку степени трудности, рисков, затрат и сформировать рабочий график	ПК-7
Способен создавать программное обеспечение для ЭВМ и систем	ПК-9
Способен читать, понимать и выделять главную идею прочитанного исходного кода, документации	ПК-11
Способен моделировать, анализировать и использовать формальные методы конструирования программного обеспечения	ПК-12
Способен оценивать временную и емкостную сложность программного обеспечения	ПК-13
Способность создавать программные интерфейсы	ПК-14
Способен использовать различные технологии разработки программного обеспечения	ПК-16
Способен применять основные методы и инструменты разработки программного обеспечения	ПК-17

#### **4 Место дисциплины в структуре образовательной программы**

Настоящая дисциплина относится к профессиональному циклу дисциплин (Б.ПЦ), его базовой части (Б.ПЦ.Б).

В соответствии с рабочим учебным планом по направлению "Программная инженерия" дисциплина "Распределённые вычисления" читается студентам четвертого курса бакалавриата в 1-ом и 2-ом модулях.

Изучение данной дисциплины базируется на знаниях, полученных студентами при освоении учебных дисциплин "Дискретная математика", "Программирование", "Архитектура вычислительных систем", "Алгоритмы и структуры данных", общих знаниях математики и основ программирования.

Дисциплина является основой для подготовки выпускной квалификационной работы подготовки бакалавра по направлению 09.03.04 "Программная инженерия".



## 5 Тематический план учебной дисциплины

№	Название темы	Всего часов по дисциплине	Аудиторные часы		Самостоятельная работа
			Лекции	Семинары	
1	Централизованные и распределённые вычисления: преимущества и недостатки.	10	2	2	6
2	Анализ предмета распределённых вычислений.	10	2	2	6
3	Вопросы взаимодействия в системах распределённых вычислений (СПВ)	12	2	2	8
4	MPI - сервис обмена сообщениями в СПВ	12	2	2	8
5	История и перспективы практической реализации распределённых вычислений.	10	2	2	6
6	WEB-компоненты как инструментальный формирование СПВ на основе модели «клиент-сервер»	14	2	2	10
7	Обзор технологий, стандартов, реализаций GRID-систем, инструментальный разработки. Правовые особенности	14	2	2	10
8	Модель «клиент-сервер» в системах распределённых	10	2	2	6
9	Объектные распределённые системы.	12	2	2	8
10	Агентные технологии в распределённых системах.	12	2	2	8
11	Компонентные системы. Концепция JavaBeans.	12	2	2	8
12	Сервис-ориентированная архитектура (COA) в распределённых системах. Концепция COA.	12	2	2	8
13	WEB-сервисы. Стандарты WSDL, SOAP. Спецификация WSRF.	14	2	2	10
14	Технология одноранговых сетей. Достоинства и недостатки P2P	12	2	2	8
15	Облачные вычисления: понятия и основные черты, многослойность архитектуры облачных приложений и их основные компоненты	12	2	2	8
16	Распространённые облачные платформы, удобства с точки зрения пользователя	12	2	2	8
<b>Итого:</b>		<b>190</b>	<b>32</b>	<b>32</b>	<b>126</b>



## 6 Формы контроля знаний студентов

Тип контроля	Форма контроля	Модули		Параметры
		1	2	
Текущий	Контрольная работа		8-я неделя	Решение задачи в сети GRID (индивидуальное задание - по указанию преподавателя).
	Оценка работы на семинаре	На каждом занятии		<ul style="list-style-type: none"><li>• подготовка сообщения;</li><li>• устное выступление;</li><li>• ответы на вопросы;</li><li>• участие в обсуждении.</li></ul>
Итоговый	Экзамен		*	Устный экзамен 60 мин.

### 6.1 Критерии оценки знаний, навыков

Оценки по всем формам контроля выставляются по 10-ти балльной шкале.

**Текущий контроль** предусматривает контрольную работу, выполняемую во 2 модуле, и оценку работы на семинарах 1 и 2 модулей.

**Контрольная работа** заключается в решении вычислительной задачи в распределённом сетевом сервисе сети GRID. Выполняется на семинарском занятии. Оценка за контрольную работу выставляется по критериям, описанным в примерном задании на контрольную работу (см. п. 9.1.1). Передача контрольной работы с целью повышения оценки не допускается. Допускается выполнение контрольной работы в более поздний срок при пропуске занятия по уважительной причине.

В рамках текущего контроля также учитывается **работа студентов на семинарских занятиях**, а именно: подготовка и выступление на семинаре в форме устного сообщения (доклада) по одной из тем семинарского занятия, участие в обсуждении сообщений, участие в тестировании по некоторым темам.

**Итоговый контроль**: экзамен в конце 2-го модуля. Проводится в устной форме. Экзамен состоит из двух частей:

- теоретической, проводится в форме устной беседы по тематике дисциплины (40 мин.);
- практической, связанной с обсуждением результатов текущего контроля (20 мин.).

### 6.2 Порядок формирования оценок по дисциплине

Оценка работы студентов на семинарских занятиях –  $O_{C3}$  – формируется путем пересчета по 10-балльной шкале арифметической суммы оценок, полученных студентом за работу на каждом семинарском занятии. Оценка работы студента на каждом семинаре оценивается по следующим критериям:

- полнота и правильность освещения темы доклада (max 5 баллов);
- качество устного выступления, владение темой (max 4 балла);
- качество презентации, сопровождающей устное выступления (1 балл);
- качество ответов на возможные вопросы по теме выступления (max 2 балла);
- активность в обсуждении докладов: количество вопросов по теме докладов, количество и качество дополнений (каждый вопрос оценивается 1 баллом, за каждое дополнение студент может получить max 2 балла);
- результаты опросов по текущей теме как устных, так и в форме компьютерного тестирования;



Выполнение контрольной работы  $O_{\text{кр}}$  оценивается по 10-бальной шкале по критериям: правильность, полнота и своевременность.

Оценки за работу на семинарских занятиях и контрольную работу выставляются в рабочую ведомость.

Накопленная оценка за текущий контроль учитывает результаты студента по текущему контролю следующим образом:

$$O_{\text{накопл.}} = 0,7 \times O_{\text{сз}} + 0,3 \times O_{\text{кр}} .$$

Результирующая (итоговая) оценка по дисциплине определяется соотношением:

$$O_{\text{итог.}} = 0,5 \times O_{\text{накопл.}} + 0,5 \times O_{\text{экс.}} ,$$

где оценка  $O_{\text{экс.}}$  – оценка за экзамен по 10-бальной шкале.

Способ округления значений при вычислениях по указанным формулам – арифметический.

Перевод в пятибалльную оценку осуществляется в соответствии со следующей таблицей.

**Таблица соответствия оценок по десятибалльной и пятибалльной системам**

<i>По десятибалльной шкале</i>	<i>По пятибалльной шкале</i>
1 – неудовлетворительно	неудовлетворительно – 2
2 – очень плохо	
3 – плохо	
4 – удовлетворительно	удовлетворительно – 3
5 – весьма удовлетворительно	
6 – хорошо	хорошо – 4
7 – очень хорошо	
8 – почти отлично	отлично – 5
9 – отлично	
10 – блестяще	



## 7 Содержание дисциплины

### 7.1 Содержание лекций

#### **Тема 1. Централизованные и распределённые вычисления: преимущества и недостатки**

Преодоление технологических ограничений вычислительной мощности процессора на прицепах распараллеливания вычислений. Соотношение понятий, «параллельные вычисления», «распределённые вычисления», «распределённая вычислительная система (РВС)». Параллелизм по операциям и параллелизм по данным, сложности разработки параллельных алгоритмов. Проблемы хранения и обработки больших и сверхбольших массивов данных. Исторические примеры разработки параллельных вычислительных систем. Практическая идентичность понятий «распределённые вычисления» и «распределённая обработка информации». Близко- и далеко- расположенные узлы обработки данных в РВС. Роль компьютерных сетей в создании РВС, использование возможностей сети InterNet в разработке РВС. Необходимые требования к разработке и функционированию РВС. Концепция прозрачности в РВС. Терминология и классификация РВС. Понятия SaaS, PaaS и IaaS в современных коммерческих РВС. Проблемность РВС с точки зрения пользователя, критика идеологии таких систем со стороны Р. Столлмана. Учебно-методическое и информационное обеспечение дисциплины «Распределённые вычисления».

#### Литература по теме 1:

1. Э.Таненбаум, М.ван Стеен. Распределённые системы. Принципы и парадигмы. – СПб., изд. Питер, 2003. – 877 с.
2. Дж.Рассел, Рон.Кон. Распределённые вычисления. – М.: Изд. VSD, 2013. – 103 с.
3. Распределённые вычисления. GRID-системы. [Электронный ресурс] – Режим доступа: <http://www.ibm.com/developerworks/ru/library/l-grid> (дата обращения: 26.10.2017).

#### **Тема 2. Анализ предмета распределённых вычислений**

Основные отличительные признаки РВС: отсутствие единого времени, отсутствие общей памяти, топологическая и географическая распределённости, независимость и гетерогенность компонентов. Цели построения РВС: формирование географически распределённой вычислительной (информационной) среды, увеличения производительности вычислений, совместное использование ресурсов, повышение отказоустойчивости. Свойства (особенности) распределённых алгоритмов: отсутствие знания глобального состояния, отсутствие общего единого времени. Систематизация и анализ основных требований к РВС: прозрачность (*transparency*), открытость (*openness*), безопасность (*security*), масштабирование (*scalability*). Технологии масштабирования: распространение (*distribution*), репликация (*replication*), кэширование (*caching*). Заблуждения относительно распределённых приложений (по П. Дейчу) и сложности разработки распределённых систем.

#### Литература по теме 2:

1. Косяков М.С. Введение в распределённые вычисления. - СПб: НИУ ИТМО, 2014. – 155 с.
2. Kshemkalyani A. D., Singhal M. Distributed Computing: Principles, Algorithms, and Systems. Cambridge University Press, 2008.
3. Э.Таненбаум, М.ван Стеен. Распределённые системы. Принципы и парадигмы. – СПб., изд. Питер, 2003. – 877 с.





### **Тема 3. Вопросы взаимодействия в системах распределённых вычислений**

Понятие и назначение программного обеспечения промежуточного уровня (*middleware*) при реализации функционала РВС. Различия между средствами взаимодействия сетевых операционных систем и программным обеспечением промежуточного уровня РВС. Проблема представления единого физического времени в РВС. Синхронные и асинхронные РВС. Упорядочивание событий при выполнении процессов в РВС. Понятия логических часов (*logical clock*) и логического времени (*logical time*) Л. Лэмпорта. Примитивы взаимодействия. Синхронный и асинхронный обмен сообщениями.

#### Литература по теме 3:

1. Косяков М.С. Введение в распределенные вычисления. – СПб: НИУ ИТМО, 2014. – 155 с.
2. Kshemkalyani A. D., Singhal M. Distributed Computing: Principles, Algorithms, and Systems. Cambridge University Press, 2008.
3. Charron-Bost B., Tel G., Mattem F. Synchronous, asynchronous, and causally ordered communication // Distributed Computing, 9(4), 1996. P.173-191.

### **Тема 4. MPI - сервис обмена сообщениями в системах распределённых вычислений**

Необходимость обмена сообщениями в РВС. История MPI (*Message Passing Interface, интерфейс передачи сообщений*) - Уильям Гроупп, Эвин Ласк и др. Проекты Chameleon, MPICH, MPICH2, LAM-MPI, WMPI, MS MPI. Принцип работы MPI - понятие менеджера процессов, его назначение. MPI как основа обеспечения объединения вычислительных узлов в кластерных системах. Независимость MPI от конкретной коммуникационной сети. Ориентированность MPI на системы с распределённой памятью, возможность совместного использования с OpenMP. Понятие коммутатора и логической топологии в MPI. Блокирующие и неблокирующие операции в MPI, ситуация взаимоблокировка (*deadlock*). Стандарт MPI-2, отличия от MPI. Стандартизация MPI посредством MPI Forum. Связь MPI с моделью акторов, лямбда-исчислением А. Чёрча и др. Минимальный набор MPI-вызовов для реализации обмена данными между процессами. Шаблон (*skeleton*) программы с использованием MPI. Пример параллельной программы умножения матриц с использованием MPI. Использование MPI как низкоуровневого инструмента при разработке РВС (Globus, SciDB, MS Windows Compute Cluster Server).

#### Литература по теме 4:

1. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. —СПб.: БХВ-Петербург, 2004. - 609 с.
2. Информационно-аналитические материалы по параллельным вычислениям. [Электронный ресурс] — Режим доступа - URL: <http://parallel.ru> (дата обращения: 26.10.2017).
3. Журнал "Суперкомпьютеры". [Электронный ресурс] — Режим доступа - URL: <http://supercomputers.ru> (дата обращения: 26.10.2017).
4. Сухинов А.А. MPICH и Windows. [Электронный ресурс] — Режим доступа - URL: <http://iprocc.ru/programming/mpich-windows> (дата обращения: 26.10.2017).

### **Тема 5. История и перспективы практической реализации распределённых вычислений**

Поколения СРВ. Первое поколение – проекты FAFNER, I-WAY (1995 г.). Второе поколение Globus, Legion, объектные распределённые системы (1997 г.). Современное состояние – одноранговые (*Peer-to-Peer, P2P*) системы, сервис-ориентированная архитектура (*SOA – Service-Oriented Architecture*), агентная архитектура, облачные вычисления. Определения Я. Фостером понятия РВС. GRID (2000 г.) и облачные вычисления (2007-2008 г.г.). Основная задача GRID, реализации. Достоинства объектного подхода в разработке РВС. Архитектура



CORBA (*Common Object Request Broker Architecture* - *общая архитектура брокера объектных запросов*). Концептуальные модели архитектуры РВС, язык UML (*Unified Modeling Language* - *Унифицированный Язык Моделирования*). Технологии RMI (*Remote Method Invocation* - *Удалённый Вызов Методов*), платформа .NET. Принцип P2P - системы Napster, Gnutella, пиринговые сети. Принципы сервис-ориентированного подхода, стандарты SOAP (основанный на XML протокол взаимодействия WEB-сервисов), WSDL (*Web Services Description Language* - *язык описания WEB-сервисов*), UDDI (*Universal Description Discovery and Integration* - *Универсальный метод поиска и интеграции*). Агентная архитектура - языки взаимодействия агентов (*Agent Communication Languages, ACLs*), архитектура взаимодействия FIPA (*Foundation for Intelligent Physical Agents* - *Основа Интеллектуальных Физических Агентов*). Проект "Blue Cloud" фирмы IBM и Европейский проект "Joint Research Initiative for Cloud Computing". Облачные вычисления как бизнес-платформа XXI века.

#### Литература по теме 5:

1. Э.Таненбаум, М.ван Стеен. Распределённые системы. Принципы и парадигмы. — С.Пб., изд. Питер, 2003. - 877 с.
2. Радченко Г.И. Распределённые вычислительные системы. — Челябинск: Изд. Фотохудожник, 2012. -184 с.: с ил.
3. С.И.Сантри, Пол Дж.Дейтел, Харви М.Дейтел. Технологии программирования на Java. Распределённые приложения. —М.: Изд. Бином-Пресс, 2009. -466 с.
4. Джеффри Рихтер. CLR via C#. Программирование на платформе Microsoft.NET Framework 4.5 на языке C#. —С.Пб., изд. Питер, 2013. -896 с.
5. Питер Фингар. DOT.CLOUD. —М.: Аквармариновая книга, 2011. -256 с.

#### **Тема 6. Обзор концептуальных компонентов WEB как распределённой системы, создание с использованием WEB-сервера приложений в сети internet на основе модели «клиент-сервер»**

История WWW (*World Wide Web*), основоположники – сэр Тимоти Джон Бернерс-Ли, Роберт Кайо. WWW как распределённая система, ориентированная на предоставление сервисов. Спецификации URI, HTML, HTTP. Всемирная компьютерная сеть internet как основа WWW. WEB как «сумма технологий» – система именования (идентификации) ресурсов, система представления ресурсов и система передача ресурсов. Понятия URI (*Uniform Resource Identifier*), URL (*Uniform Resource Locator*), URN (*Uniform Resource Name*). Принципы формирования доменных имён в WEB. IP-адресация и её версии, причины перехода на 6-ю версию IP. Служба доменных имён (*Domain Name Service, DNS*). Бернерс-Ли и «принцип наименьшей силы» (*the Principle of Least Power*) при описании ресурсов. Язык описания гипертекста HTML, единый протокол передачи гипертекста HTTP. Поиск информации в WEB как сервис, поиск нужной информации (*Data Mining*). Принципы работы поисковых систем. Закон Зипфа в применении к поиску информации и правила формирования ключевых слов. Понятие WEB-сервера как специализированного программного обеспечения для создания сервисов в WEB (*SaaS - Software-as-a-Service, программное обеспечение как услуга для потребителей*). Системные основы расширения функциональности WEB-серверов.: механизм взаимодействия процесса-родителям (*parent process*) и процессов-потомков (*spawn process*), системные вызовы реализации этого взаимодействия. Технологии CGI (*Common Gateway Interface*) и ISAPI (*Internet Server Application Program Interface*). Формы (*forms*) в HTML, методы Get и Post. Расшифровка процессом-потомком сервера строки сообщения, выполнение требуемой функциональности и возврат результата выполнения запроса клиентской части. Пример: формирование SQL-запроса по данным разбора (*parsing*) передаваемой клиентом строки. Программные примеры реализации технологии CGI (счётчик числа посещений HTML-страницы). Парадигма клиент-серверной архитектуры. Протоколы систем «клиент–сервер» (C/S) – без установки соединения и с установкой соединения, область их применения,



достоинства и недостатки. Три уровня функциональности в системах C/S: уровень представления (*пользовательского интерфейса*), уровень обработки (*бизнес-логики*), уровень работы с данными. Классическая двухзвенная архитектура C/S, её недостатки. Понятие «толстого» и «тонкого» клиентов. Трёхзвенная C/S архитектура, функции сервера приложений, достоинства по сравнению с двухзвенной архитектурой. Пример реализации двухзвенной C/S архитектуры – поисковая машина в сети InterNet.

#### Литература по теме 6:

1. Э.Таненбаум, М.ван Стеен. Распределённые системы. Принципы и парадигмы. —СПб., изд. Питер, 2003. - 877 с.
2. Радченко Г.И. Распределённые вычислительные системы. —Челябинск: Изд. Фотохудожник, 2012. -184 с.: с ил.
3. Леон Шкляр, Рич Розен. Архитектура веб-приложений. —М.: Серия: Высший уровень. Изд. Эксмо, 2011. - 640 с.
4. Попов Артём. Эффективная методика поиска информации в сети Интернет. [Электронный ресурс] — Режим доступа - URL: [http://citforum.ru/pp/search\\_03.shtml](http://citforum.ru/pp/search_03.shtml) (дата обращения: 26.10.2017).
5. Жерар Тель. Введение в распределённые алгоритмы. — М.: Изд. МЦНМО, 2009. -616 с.
6. Дж.Рассел, Рон.Кон. Распределённые вычисления. — М.: Изд. VSD, 2013. -103 с.
5. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. — СПб.: БХВ-Петербург, 2004. — 609 с.

#### **Тема 7. Обзор технологий, стандартов, реализаций GRID-систем, инструментарий разработки. Правовые особенности использования GRID**

Определение GRID (Грид), происхождение термина. Понятия GRID и метакомпьютера (метакомпьютеринга). Основная концепция и технологии построения GRID. Функциональность уровней архитектуры GRID. Стандарты GRID. Технология WEB-сервисов как база стандарта GRID-приложений. Стандарт GRID-архитектуры OGSA (*Open Grid Services Architectue - Открытая архитектура Грид-сервисов*). Стандартные интерфейсы GRID-сервиса. Реализации модели OGSA. Система Globus: цель разработки, основная направленность, укрупнённая схема взаимодействия компонентов и базовые сервисы системы Globus. Система UNICORE (*Uniform Interface to Computing Resources - Единый интерфейс к вычислительным ресурсам*): архитектура, программные слои. Программная платформа BOINC (*Berkeley Open Infrastructure for Network Computing*), проект VCSC (*Create a Virtual Campus Supercomputing Center - Создание виртуального суперкомпьютерного центра кампуса*). Модели производительности.

Ограничения на задачи, эффективно решаемые с помощью GRID. Потенциал вычислительных мощностей GRID. Сравнение GRID и суперкомпьютеров. Проблемы гетерогенности и динамического характера конфигурации метакомпьютера. Области науки и техники, использующие GRID-технологии. Проекты EU Data GRID (EDG, проект CERN), X-Com (НИВЦ МГУ). Проект SETI@home (*Search for Extra Terrestrial Intelligence*), количественные параметры проблемы. Проекты Einstein@Home, Folding@Home, GIMPS, SeventeenOrBust, ZetaGrid и др. Достигнутые успехи в реализации GRID-проектов. Условия предоставления мощностей ПЭВМ проектам GRID. Понятие "добровольных вычислений" (*Volunteer computing*), привлечение и мотивация участников. Критика проектов распределённых вычислений. GRID и облачные технологии.

#### Литература по теме 7:

1. Э.Таненбаум, М.ван Стеен. Распределённые системы. Принципы и парадигмы. — СПб., изд. Питер, 2003. -877 с.



2. Радченко Г.И. Распределённые вычислительные системы. — Челябинск: Изд. Фотохудожник, 2012. -184 с.: с ил.
3. GRID. [Электронный ресурс] — Режим доступа - URL: <http://www.seobuilding.ru/wiki/%D0%93%D1%80%D0%B8%D0%B4> (дата обращения: 05.10.2017).
4. Российские распределённые вычисления на платформе BOINC. [Электронный ресурс] — Режим доступа - URL: <http://www.boinc.ru> (дата обращения: 05.10.2017).
5. Семенов Ю.А. Сети GRID. [Электронный ресурс] – Режим доступа URL: <http://book.itep.ru/4/7/grid.htm> (дата обращения: 05.10.2017).
6. Распределённые вычисления: волонтеры на службе науки (интернет - журнал "Распределённые вычисления"). [Электронный ресурс] — Режим доступа - URL: <http://www.vechnayamolodost.ru/pages/drugienukiozhizni/rasvyvonaslnab6.html> (дата обращения: 26.10.2017).

### **Тема 8. Модель «клиент-сервер» в системах распределённых вычислений**

Парадигма клиент-серверной архитектуры. Терминология систем клиент-сервер. Протоколы систем "клиент - сервер" (C/S) - без установки соединения и с установкой соединения, область их применения, достоинства и недостатки. Три уровня функциональности в системах C/S: уровень представления (пользовательского интерфейса), уровень обработки (бизнес-логики), уровень работы с данными. Функциональность каждого уровня, физическое расположение программ-обработчиков каждого уровня. Концепция независимости данных от приложений. Реляционная база данных (БД) концепция отношений. Представление данных в понятиях объектов, объектно-ориентированные БД, миграция функциональности с уровня обработки на уровень данных. Классическая двухзвенная архитектура C/S, её недостатки. Понятие "толстого" и "тонкого" клиентов. Трёхзвенная C/S архитектура, функции сервера приложений, достоинства по сравнению с двухзвенной архитектурой. Пример реализации двухзвенной C/S архитектуры - поисковая машина в сети Internet. Методы горизонтального и вертикального распределения приложений и контента БД, приёмы выравнивания нагрузки на сервера. Концепция облачных вычислений как пример горизонтального клиент-серверного распределения.

#### Литература по теме 8:

7. Э.Таненбаум, М.Вай Стеен. Распределённые системы. Принципы и парадигмы. С.Пб.: изд. Питер, 2003.
8. Леон Шкляр, Рич Розен. Архитектура веб-приложений. М.: Серия: Высший уровень. Изд. Эксмо, 2011.
9. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. СПб.: БХВ-Петербург, 2004.
10. Вычислительные сети. [Электронный ресурс] — Режим доступа - URL: <http://lanhelper.ru> (дата обращения: 26.10.2017).

### **Тема 9. Объектные распределённые системы**

Объектно-ориентированный подход к созданию систем распределённых вычислений. Поддержка абстракции данных, инкапсуляции, наследования и полиморфизма. Сериализация и десериализация данных. Технологии вызова удалённых процедур (RPC, *Remote Procedure Call*), технология CORBA (*Common Object Request Broker Architecture - Обобщённая архитектура брокера объектных запросов*), общая технология использования объектно-ориентированных подходов к проектированию РВС. RPC - парадигма, решение в конкретных случаях. Понятия "заглушки" (клиентского и серверного "стаба", *stub*), функция "заглушки". перемещение информации на клиентской и серверной сторонах при вызове "заглушки". Общая концепция организация связи с удалённым объектом. Процедуры сериализации и десериализации



объекта. Технология Java RMI (*Remote Method Invocation, вызов удалённых методов*), достоинства и недостатки. Порядок выполнения действий на стороне клиента и сервера при RMI. Технология разработки РВС-приложений CORBA (*Common Object Request Broker Architecture, - обобщенная архитектура брокера - посредника - объектных запросов*). Консорциум OMG (*Object Management Group*), инфраструктура *Object Management Architecture* (OMA), язык определения интерфейсов распределённых объектов IDL (*Interface Definition Language*), компонент ORB (*Object Resource Broker - брокер ресурсов объектов*). Спецификация CORBA 2.0. Разработки на основе CORBA, недостатки CORBA.

#### Литература по теме 9:

1. Э.Таненбаум, М.ван Стеен. Распределённые системы. Принципы и парадигмы. —СПб., изд. Питер, 2003. -877 с.
2. С.И.Сантри, Пол Дж.Дейтел, Харви М.Дейтел. Технологии программирования на Java. Распределённые приложения. —М.: Изд. Бином-Пресс, 2009. -466 с.
3. Радченко Г.И. Распределённые вычислительные системы. — Челябинск: Изд. Фотохудожник, 2012. -184 с.: с ил.

### **Тема 10. Агентные технологии в распределённых системах**

Определение программного агента. Задачи, решаемые с помощью агентных технологий. Автономные агенты. Основные составляющие автономного агента. Правила функционирования простого автономного агента. Свойства автономного агента. Свойства реактивности, автономности, целенаправленности, коммуникативности. Обучаемые агенты. Понятие интеллектуального агента. Регулирующие и планирующие агенты. Мультиагентные системы, отличие мультиагентной системы от одноагентной. Понятия агентной платформы и агентной системы. Распространённые агентные платформы - JADE (*Java Agent DEvelopment Framework*), Cougaar (*Cognitive Agent Architecture*). Вопросы безопасности в системах мобильных агентов. Анализ конкретных проблем безопасности и возможные решения. Решения проблемы безопасности агента при атаках со стороны исполняющей среды («хоста»).

#### Литература по теме 10:

1. Ком Э.Таненбаум, М.ван Стеен. Распределённые системы. Принципы и парадигмы. — СПб., изд. Питер, 2003. -877 с.
2. JADE - Java Agent DEvelopment Framework. [Электронный ресурс] — Режим доступа - URL: <http://jade.tilab.com> (дата обращения: 05.10.2017).

### **Тема 11. Компонентные системы. Концепция JavaBeans**

Компонентно-ориентированный подход (КОП) как развитие объектно-ориентированного подхода. Практическая направленность КОП на разработку крупных и распределённых программных систем. Представление программной системы со стороны КОП, отличие от иных подходов программной инженерии (напр., от *рефакторинга* - постепенного изменения внутренней структуры программы без изменения её внешнего поведения с целью облегчения понимания работы программы). Понятие программного компонента, отличие его от объекта в смысле ООП. Открытый интерфейс, механизм взаимодействия с программной средой. Основные преимущества применения КОП. Требования к разработке программных компонентов. Концепция JavaBeans, цель JavaBeans, спецификация Sun Microsystems. Правила (соглашения) описания JavaBeans. Спецификация, инфраструктура и технология Enterprise JavaBeans (EJB). Решение задач масштабирования серверов приложений с использованием Java. Компоненты EJB, реализация компонентов в виде удалённого интерфейса, собственного интерфейса и реализации. Параметры развёртывания (*deployment*) наборов компонент, контейнер EJB, предназначение контейнера EJB, сервер EJB. Типы компонентов EJB -



сессии (*session*) и сущностные (*entity*). Составные части EJB-компонента. Инфраструктура Enterprise JavaBeans. Java Application Programming Interface, отсутствие связи с реализацией.

Литература по теме 11:

1. Э.Таненбаум, М.ван Стеен. Распределённые системы. Принципы и парадигмы. – СПб., изд. Питер, 2003. -877 с.
2. Радченко Г.И. Распределённые вычислительные системы. – Челябинск: Изд. Фотохудожник, 2012. -184 с.: с ил.
3. Введение в технологию Enterprise JavaBeans. [Электронный ресурс] – Режим доступа – URL: <http://citforum.ru/internet/javascript/enterpjvabeans.shtml> (дата обращения: 05.10.2017).

**Тема 12. Сервис-ориентированная архитектура в распределённых системах. Концепция СОА**

Определение сервис-ориентированной архитектуры (СОА, *Service-Oriented Architecture - SOA*), основа СОА. Международная организация по стандартам и спецификациям (*OASIS - Organization for the Advancement of Structured Information Standart*). Составляющие СОА - сервисные компоненты (собственно *сервисы*), интерфейсы (*контракты* сервисов), соединители (*транспорты* сервисов), механизмы обнаружения (*регистры* сервисов). WEB-сервисы как наиболее распространённые реализации СОА. Независимость WEB-сервисов от платформы и языка программирования. Связанность программных средств, сильно- и слабосвязанные системы. Принципы построения СОА. Понятие интероперабельности - техническая и семантическая интероперабельность. Принципы проектирования СОА. Логика предприятия с точки зрения подхода СОА - бизнес-логика и логика приложения. Понятие *СааS (Cognition-as-a-Service - распознавание как сервис)*. Рекомендуемые принципы построения сервисов.

Литература по теме 11:

1. Э.Таненбаум, М.ван Стеен. Распределённые системы. Принципы и парадигмы. – СПб., изд. Питер, 2003. -877 с.
2. Advancing open standards for the information society. [Электронный ресурс] – Режим доступа - URL: <https://www.oasis-open.org> (дата обращения: 05.10.2017).
3. Сервис-ориентированная архитектура. [Электронный ресурс] – Режим доступа - URL: <http://citforum.ru/internet/webservice/soa> (дата обращения: 05.10.2017).

**Тема 13. WEB-сервисы. Стандарты WSDL, SOAP. Спецификация WSRF**

Определение WEB-сервисов. Поколения WEB-сервисов. Спецификация стандартов взаимодействия WEB-сервисов - протокол SOAP (обмен сообщениями), WSDL (*Web Services Description Language* - описание сервисов в виде абстрактных ресурсов), UDDI (*Universal Description Discover & Integration* - стандарт внутреннего устройства и внешнего интерфейса репозитория описаний интерфейсов). Представление сервиса в WSDL с абстрактной и конкретной точек зрения. Спецификации WSDL 2.0 - WSDL Part 1: Core Language (*Основной язык*), WSDL Part 2: Message exchange patterns (*Шаблоны обмена сообщениями*) и WSDL Part3: Binding (*Связывание*). Стэк протоколов WEB-сервисов. Пример простого сервиса с использованием Java. Формирование WSDL-документа. Понятия *портов, сообщений, связей, блоков* WSDL. Стандарт WSDL 2.0, изменения и дополнения по сравнению с предыдущим. Стандарт SOAP: понятия *конверта, заголовка* и тела *сообщения*. Пример заголовка, шаблона, запроса/ответа SOAP-сообщения. Паттерн поведения "запрос/ответ" при связывании SOAP и HTTP- протокола. Второе поколение WEB-сервисов, безопасность WEB-сервисов и стандарт обеспечения безопасности WS-Security. Варианты обеспечения авторизации. Подпись сообщения, шифрование путём XML Encryption. Адресация, стандарт WS-Addressing. Понятие



состояния WEB-сервисов и спецификации WSRF (*Web Services Resource Framework*). Примеры описаний WS-ресурсов и операций с ними, запросы на создание и получение информации о состоянии WS-ресурса.

#### Литература по теме 13:

1. Э.Таненбаум, М.ван Стеен. Распределённые системы. Принципы и парадигмы. – СПб., изд. Питер, 2003. -877 с.
2. Windows Wev Services. [Электронный ресурс] – Режим доступа - IRL: <http://msdn.microsoft.com/ru-ru/magazine/ee335693.aspx> (дата обращения: 05.10.2017).
3. Радченко Г.И. Распределённые вычислительные системы. – Челябинск: Изд. Фотохудожник, 2012. -184 с.: с ил.

#### **Тема 14. Технология одноранговых сетей. Достоинства и недостатки P2P**

Принципы функционирования одноранговых (P2P) сетей. Сравнение 323 и централизованной (клиент-серверной) архитектур. Задачи P2P-сетей. Основные понятия и терминология P2P- сетей. Протоколы сетей архитектуры P2P. Структура P2P-сетей. Алгоритмы функционирования P2P-сетей. Применение технологии P2P: распределённые вычисления, файлообменные сети, приложения для обеспечения совместной работы. Платформа BOING. Система BitTorrent. Технологии "магнитных ссылок" (*magnet links*) и распределённых хеш-таблиц (DHT - *Distributed Hash Table*). Internet-служба Skype. Достоинства и недостатки P2P-архитектуры.

#### Литература по теме 14:

1. Э.Таненбаум, М.ван Стеен. Распределённые системы. Принципы и парадигмы. —СПб., изд. Питер, 2003. -877 с.
2. P2P социальная сеть Pandora. [Электронный ресурс] — Режим доступа - URL: <http://habrahabr.ru/post/164149/> (дата обращения: 05.10.2017).

#### **Тема 15. Облачные вычисления: понятия и основные черты, многослойность архитектуры облачных приложений и их основные компоненты**

Метафора "Cloud" (Облако). Начало бурного развития облачных вычислений. Преимущества облачных технологий с точки зрения бизнеса. Формальные определения облачных вычислений. Понятие "коммунальных вычислений" (*Utility Compuing*). Виртуализация и масштабируемость облачных вычислений. Трёхслойная архитектура облачных систем (IaaS, PaaS, SaaS), принципы взаимодействия слоёв при облачных вычислениях. Инфраструктура как сервис (*Infrastructure as a Service - IaaS*). Платформа как сервис (*Platform as a Service - PaaS*), понятие платформы как слоя абстракции между программными приложениями (SaaS) и виртуализированной инфраструктурой (IaaS), круг пользователей PaaS. Программное обеспечение как сервис (*Software as a Service - SaaS*), принцип предоставления SaaS- услуг. Компоненты облачных приложений - платформа, представление, информация, идентификация, интеграция, масштабируемость, монетизация, внедрение, функционирование. Типы потребителей облачных сервисов. Достоинства облачных вычислений с точки зрения экономики, простоты доступа и надёжности использования ресурсов. Проблемы безопасности и конфиденциальности информации при использовании "облачного аутсорсинга". Классификация облаков. Общественные и частные облака. Гибридные облака и федерации облаков.

### Литература по теме 15:

1. Радченко Г.И. Распределённые вычислительные системы. – Челябинск: Изд. Фотохудожник, 2012. -184 с.: с ил.
2. Сафонов В.О. Платформа облачных вычислений Microsoft Windows Azure. Учебное пособие. — М.: Национальный открытый университет "ИНТУИТ": БИНОМ. Лаборатория знаний, 2013. - 234 с.
3. Риз Дж. Облачные вычисления. – СПб.: БХВ-Петербург, 2011. - 288 с.

### **Тема 16. Распространённые облачные платформы, удобства с точки зрения пользователя**

Современные наиболее распространённые облачные платформы. Облачная платформа Amazon как фактический стандарт в мире облачных технологий. Понятие виртуальных ресурсов (виртуальных машин), возможность использования собственных решений (расширяемость платформы). Прозрачность адресации и доступность в сервисах интеграции приложений, сервис виртуальных частных облаков Amazon. Платформа Google App Engine, ориентация на PaaS. Используемые языки программирования, понятие "песочницы", предоставление разработчику полноценного SDK. Хранилище данных App Engine Datastore, язык запросов к БД GQL. Обработка данных посредством MapReduce. Платформа MS Windows Azure, ориентированность на MS .NET, возможность тестирования приложений до их опубликования в Windows Azure. Удобство разработки приложений с использованием Visual Studio и различных языков программирования. Компоненты MS Windows Azure - вычислительные сущности, сущности хранения, "фабрики". Поддержка балансировки нагрузок. Понятие роли, два типа ролей (WEB-роль - Web Role и прикладная роль - Worker Role) в MS Windows Azure.

### Литература по теме 16:

1. Питер Фингар. DOT.CLOUD. М.: Акварининовая книга, 2011.
2. Федоров А.Г., Мартынов Д.Н. Windows Azure: облачная платформа Microsoft. [Электронный ресурс] — Режим доступа - [URL:http://download.microsoft.com/documentcs/rus/msda/Windows\\_Azure\\_wcb.pdf](http://download.microsoft.com/documentcs/rus/msda/Windows_Azure_wcb.pdf)
3. Риз Дж. Облачные вычисления. СПб.: БХВ-Петербург, 2011.

## **7.2 Содержание семинарских занятий**

Темы семинарских занятий определяются тематическим планом лекционных занятий (см. 7.1).

1. Углубленное изучение специальных вопросов по темам лекций (с использованием информационных материалов преподавателя и/или ресурсов сети InterNet).
2. Выступление студентов с подготовленным сообщением; сообщение сопровождается демонстрацией презентации. ;
3. проводится обсуждение
4. Изучение целей и особенностей функционирования известных GRID-приложений (SETI@home, Einstein@Home, Folding@Home и др.)

Языком программной реализации алгоритмов является язык программирования C++/C#, в связи с этим на занятиях повторяются основы языка C++/C## в объёме, необходимом для выполнения практических заданий.





## 8 Образовательные технологии

Участие в семинарских занятиях предполагает подробное изучение материала по текущей теме лекции (с использованием сети InterNet и материалов преподавателя); выступление на семинаре с подготовленным сообщением, участие в обсуждении сообщений, сделанных студентами, а также практическое ознакомление с технологиями GRID-вычислений на персональных компьютерах.

## 9 Оценочные средства для текущего контроля и аттестации студента

### 9.1 Тематика заданий текущего контроля

#### 9.1.1 Контрольная работа

**При выполнении задания контрольной работы студенту требуется:**

- 1) Выяснить правовые возможности использования сетевых GRID-приложений на данном персональном компьютере (в соответствии с правилами *добровольных вычислений* такие приложения можно запускать только на компьютерах, которыми *данное лицо владеет* или для которых *получено право владельца*; в некоторых учреждениях запрещается использовать компьютеры в подобных проектах).
- 2) Найти в сети Internet заданное преподавателем приложение класса GRID, установить его на персональном компьютере и добиться устойчивой работы приложения.
- 3) Провести цикл расчётов, зафиксировать интенсивность использования ресурсов персонального компьютера и обмена информацией по сети; заархивировать необходимые данные.
- 4) Деинсталлировать установленное приложение (см. п.2)), убедиться в полном восстановлении состояния персонального компьютера, идентичному бывшему до работы с данным приложением.

#### **Формат входных данных:**

1. Конкретное развёртываемое GRID-приложение задаётся преподавателем индивидуально. Дополнительно выдаётся задание по анализу функционирования приложения на клиентской машине, напр.:
  - определить режим использования распределённым приложением ресурсов клиентской машины (работа в фоновом режиме, в "режиме скринсейвера", процент использования вычислительных мощностей, дисковой памяти)
  - оценить общее количество информации, переданное при функционировании приложения в обе стороны по сети (трафик сети)
  - сделать вывод об использовании графической карты (если имеется) для вычислений.
2. При анализе режима использования ресурсов и сетевого трафика преподавателем может быть указан инструмент измерения (напр., Диспетчер Задач для MS Windows или встроенный в клиент измеритель трафика). Параметры усредняются на заданном промежутке времени.

#### **Формат выходных данных**

Выходные данные выдаются в свободном формате, при оценке количественных характеристик конкретизируются единицы измерения (напр., скорость обмена данным по сети - в байт/сек, количество переданной информации - Байт или килоБайт).



### Пример входных и выходных данных

<i>Входные данные (задание)</i>	<i>Выходные данные (результаты)</i>
<ol style="list-style-type: none"><li>1. Инсталлировать и проанализировать работу клиентского приложения проекта распределённых вычислений SETI@home</li><li>2. Проанализировать использование ресурсов персонального компьютера и сетевого трафика на протяжении 60 мин.</li></ol>	<ol style="list-style-type: none"><li>1. Найден сервер проекта <a href="http://setiathome.berkeley.edu">http://setiathome.berkeley.edu</a></li><li>2. Клиентская часть (BOING Manager, размер загружаемого файла 9,48 Мбайт) в версии 7.2.23 для операционной системы Windows 64 бит загружена с адреса <a href="http://boinc.berkeley.edu/download.php">http://boinc.berkeley.edu/download.php</a> (время 13.30)</li><li>3. Клиент SETI@home стартовал в 13.40 в режиме скринсейвера, графическая карта NVIDIA на персональном компьютере используется для расчётов</li><li>4. Обмен данным идёт блоками по 2 кбайт от сервера к клиенту.</li><li>5. За 60 мин эксплуатации общий объём данных от сервера к клиенту около 2,6 МБайт; от клиента к серверу - 450кБайт</li><li>6. Работа приостановлена в 14:40.</li></ol>

### Оценивание контрольной работы

- 1) Верно выполняется тестовый пример, приведённый выше - 3 балла.
- 2) Отсутствуют синтаксические ошибки в заданном примере - 4 балла.
- 3) Заданный пример выполняется не на всех входных данных - (+ до 2 баллов) (6-7 баллов).
- 4) Заданный пример выполняется на всех входных данных - (+ до 2 баллов) (8-9 баллов).

### 9.1.2 Семинарские занятия

Темы сообщений, предлагаемые для обсуждения на семинаре предстоящей недели, доводятся до студентов в заключительной части лекции. Студент фиксирует свое участие в предстоящем семинаре непосредственно после окончания лекции или по электронной почте. Типовое количество сообщений, предусмотренных для проведения семинара, - 5. Допускается подготовка сообщения на одну тему двумя студентами.

По согласованию с преподавателем студент может сделать доклад по самостоятельно сформулированному вопросу в рамках рассматриваемой темы семинара.

### 9.2 Вопросы для оценки качества освоения дисциплины

1. Понятие распределённых вычислений. Централизованные вычисления vs распределённых. Преимущества и недостатки.
2. Терминология распределённых вычислительных систем (РВС).
3. Классификация РВС, параметры классификации.
4. Связь между компонентами РВС. Понятие сетевого протокола, стека протоколов. Сетевая модель OSI.
5. Особенности РВС. Гетерогенность, технологии обработки информации в гетерогенной среде.
6. Понятие и роль промежуточного программного обеспечения в РВС.
7. История развития РВС, поколения РВС.
8. Первое поколение РВС, особенности реализации и функциональность.
9. Второе поколение РВС, особенности реализации и функциональность.
10. Современные РВС, реализации и функциональность.
11. Персоналии в области разработки РВС.
12. WEB - история развития, основные положения.
13. Сеть InterNet - развитие, основные концепции, будущее.



14. Сеть InterNet - технологии обеспечения функционирования в гетерогенной среде.
15. Методы общего представления ресурсов в сети InterNet.
16. Протокол передачи гипертекста в сети InterNet.
17. Клиент-серверная (К/С) модель РВС. Определение, достоинства и недостатки.
18. Распределение приложений по уровням в К/С-модели.
19. Варианты К/С модели. Двух- и трёхзвенные схемы.
20. Вертикальное и горизонтальное распределение в многозвенных К/С-приложений. Принципы балансировки загрузки.
21. Поиск информации в WEB как сервис. Принцип выявления ключевых слов.
22. Объектные распределённые системы. Технология RPC (*Вызовудалённых процедур*)
23. Объектные распределённые системы. Технология CORBA (*Обобщённая архитектура брокера объектных запросов*).
24. Процесс разработки приложения с использованием технологии CORBA. Этапы определения интерфейса объекта, создание кода заглушки и скелетона, разработка кода реализации и кода использования объекта. Язык определения IDL.
25. Базовые операции технологии RPC. Понятие стаба (stub), стаб клиента и стаб сервера, их совместное функционирование.
26. Организация связи с использованием удалённых объектов. Сериализация и десериализация объектов. Технология RMI.
27. Агентные технологии РВС. Понятие программного агента, его составляющие, свойства.
28. Агентные технологии РВС. Автономные и многоагентные системы. Интеллектуальный агент. Безопасность в системах с мобильными агентами.
29. Компонентные системы распределённых вычислений. Принцип компонентных систем, концепция JavaBeans.
30. Сервис-ориентированная архитектура (COA) в РВС. Понятия слоёв - IaaS, PaaS, SaaS. Принципы взаимодействия слоёв друг с другом.
31. Сервис-ориентированная архитектура (COA) РВС. Основные концепции COA, понятие связанности программных систем, принципы построения COA, требования к сервисам.
32. WEB-сервисы. Поколения WEB-сервисов, отличительные особенности. Понятие состояния WEB-сервисов, спецификация WSRF и её стандарты.
33. WEB-сервисы. Принципы адресации WEB-сервиса. Стандарт WS-Addressing.
34. Одноранговые сети (P2P). Основы архитектуры, достоинства и недостатки.
35. Одноранговые сети (P2P). Алгоритмы работы, достоинства и недостатки.
36. Одноранговые сети (P2P). Применение технологии, история развития, современное состояние.
37. Технология GRID (Грид). Направленность GRID. Архитектура GRID, стандарты GRID.
38. Технология GRID (Грид). Инструментальные средства GRID. Известные реализации GRID-сетей.
39. Технология GRID (Грид). Менеджер выполнения GRID-приложений BOINC. Проект VCSC.
40. Технология GRID (Грид). Понятие добровольности вычислений. Известные GRID- проекты.
41. Параметрические модели производительности GRID. Метрики, зависящие от времени и от объёма работы.
42. GRID и Облачные вычисления. Сравнение - общие принципы и разница.
43. Облачные приложения. Понятие виртуализации, механизмы виртуализации.
44. Облачные вычисления. Понятие облачных вычислений, многослойность архитектуры облачных вычислений. Нацеленность отдельных слоёв на конкретные типы потребителей.
45. Облачные вычисления. Классификация облаков, параметры классификации. Нацеленность на конкретных потребителей ресурсов.
46. Облачная платформа Microsoft Windows Azure. История возникновения и развития. Основные компоненты и сервисы платформы. Типы потребителей платформы. Связь с иными разработками Microsoft Corp.



47. Принципы разработки облачных приложений в среде Microsoft Visual Studio. Выбор WEB-роли, создание ASP.NET-страницы, локальная отладка, публикация и развёртывание приложения в облаке, отслеживание работоспособности приложения в облаке.
48. Облачная платформа Amazon. История возникновения и развития. Основные компоненты и сервисы платформы. Типы потребителей платформы.
49. Облачная платформа Google AppEngine. История возникновения и развития. Основные компоненты и сервисы платформы. Типы потребителей платформы.
50. Достоинства и недостатки облачных вычислений. Риски при использовании "облачного аутсорсинга".

## 10 Учебно-методическое и информационное обеспечение дисциплины

### 10.1 Основная литература - отсутствует

### 10.2 Дополнительная литература и источники

1. Э.Таненбаум, М.ван Стеен. Распределённые системы. Принципы и парадигмы. —СПб., изд. Питер, 2003. -877 с.
2. Дж.Рассел, Рон.Кон. Распределённые вычисления. —М.: Изд. VSD, 2013. -103 с.
3. Жерар Тель. Введение в распределённые алгоритмы. —М.: Изд. МЦНМО, 2009. -616 с.
4. Сафонов В.О. Платформа облачных вычислений Microsoft Windows Azure. Учебное пособие. —М.: Национальный открытый университет "ИНТУИТ": БИНОМ. Лаборатория знаний, 2013. - 234 с.
5. С.И.Сантри, Пол Дж.Дейтел, Харви М.Дейтел. Технологии программирования на Java. Распределённые приложения. —М.: Изд. Бином-Пресс, 2009. -466 с.
6. Камерон Хьюз, Трейси Хьюз. Параллельное и распределённое программирование с использованием C++. —М.: Изд. Вильямс, 2004. -672 с.
7. Джеффри Рихтер. CLR via C#. Программирование на платформе Microsoft.NET Framework 4.5 на языке C#. —СПб., изд. Питер, 2013. -896 с.
8. Радченко Г.И. Распределённые вычислительные системы. —Челябинск: Изд. Фотохудожник, 2012. -184 с.: с ил.
9. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. — СПб.: БХВ-Петербург, 2004. - 609 с.
10. Риз Дж. Облачные вычисления. —СПб.: БХВ-Петербург, 2011. - 288 с.
11. Леон Шкляр, Рич Розен. Архитектура веб-приложений. — М.: Серия: Высший уровень. Изд. Эксмо, 2011. -640 с.

### 10.3 Справочники, словари, энциклопедии

1. Распределённые вычисления. [Электронный ресурс] — Режим доступа - URL: <https://ru.wikipedia.org/wiki/%D0%A0%D0%B0%D1%81%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D1%91%D0%BD%D0%BD%D1%8B%D0%B5%D0%B2%D1%8B%D1%87%D0%B8%D1%81%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F> (дата обращения: 25.10.2017).
2. Распределённые вычисления: волонтеры на службе науки (интернет - журнал "Распределённые вычисления"). [Электронный ресурс] — Режим доступа - URL: <http://www.vechnayamolodost.ru/articles/poplem/rasvyvonaslnab6/> (дата обращения: 25.10.2017).
3. Операционные системы распределённых вычислительных систем (распределённые ОС). [Электронный ресурс] — Режим доступа - URL: <http://www.parallel.ru/krukov> (дата обращения: 25.10.2017).
4. Информационно-аналитические материалы по параллельным вычислениям. [Электронный



ресурс] — Режим доступа - URL: <http://parallel.ru> (дата обращения: 25.10.2017).

5. Журнал "Суперкомпьютеры". [Электронный ресурс] — Режим доступа - URL: <http://supercomputers.ru> (дата обращения: 25.10.2017).
6. JADE - Java Agent Development Framework. [Электронный ресурс] — Режим доступа - URL: <http://jade.tilab.com> (дата обращения: 25.10.2017).
7. Cougaar - Cognitive Agent Architecture. [Электронный ресурс] — Режим доступа - URL: <http://cougaar.org> (дата обращения: 25.10.2017).
8. Введение в технологию Enterprise JavaBeans. [Электронный ресурс] — Режим доступа - URL: <http://citforum.ru/internet/javascript/enterpjavabeans.shtml> (дата обращения: 25.10.2017).

#### **10.4 Программные средства**

Контрольная работа проводится в компьютерном классе с выходом в Интернет и доступом к ресурсам электронной библиотеки НИУ ВШЭ. Рабочее место за персональным компьютером предоставляется каждому студенту индивидуально.

1. Microsoft Visual Studio 2010.
2. Microsoft Office Professional 2007-2010.
3. Предустановленный WEB-браузер (напр., Microsoft Internet Explorer последней версии).

#### **10.5 Дистанционная поддержка дисциплины**

Дистанционная поддержка дисциплины обеспечивается использованием LMS. В разделе дисциплины "Распределённые вычисления" размещаются материалы лекций и практических занятий, тесты для самоподготовки, проекты, оценки текущего и итогового контроля.

#### **11 Материально-техническое обеспечение дисциплины**

Проектор для лекций и семинаров, классы для семинаров с компьютерами, на которых установлена инструментальная среда Microsoft Visual Studio 2010.