Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Максимов Алексей Борисович
Должность: директор департамента по образовательной политике
Дата подписания: 16.07.2025 15:47:15
Уникальный программный ключ:
8db180d1a3f02ac9e60521a5672742735c18b1d6

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение
высшего образования
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
Факультет информационных технологий

УТВЕРЖДЕНО
Декан факультета
Информационных технологий
_____/ Д.Г. Демидов /
« 27 » февраля 2025 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Теоретические основы информатики

Направление подготовки/специальность
09.03.03 Прикладная информатика

Профиль/специализация
Корпоративные решения на платформе 1С

Квалификация
бакалавр

Формы обучения
очная

Москва, 2025 г.

Разработчик(и):

Старший преподаватель  / М.В. Шульга/

Согласовано:
Заведующий кафедрой «Прикладная информатика»,
К.э.н, доцент / С.В. Суворов /



Содержание

Оглавление

1 Цели, задачи и планируемые результаты обучения по дисциплине 4
2 Место дисциплины в структуре образовательной программы 4
3 Структура и содержание дисциплины 4
3.1 Виды учебной работы и трудоемкость 5
3.2 Тематический план изучения дисциплины ..... **....5**
3.3 Содержание дисциплины 6
3.4 Тематика семинарских/практических и лабораторных занятий 7
3.5 Тематика курсовых проектов (курсовых работ) 7
4 Учебно-методическое и информационное обеспечение 7
4.1 Нормативные документы и ГОСТы 7
4.2 Основная литература 7
4.3 Дополнительная литература 8
4.4 Электронные образовательные ресурсы 8
4.5 Лицензионное и свободно распространяемое программное обеспечение 8
4.6 Современные профессиональные базы данных и информационные справочные системы 8
5 Материально-техническое обеспечение 8
5.1 Требования к оборудованию и помещению для занятий 8
5.2 Требования к программному обеспечению 8
6 Методические рекомендации 9
6.1 Методические рекомендации для преподавателя по организации обучения 9
6.2 Методические указания для обучающихся по освоению дисциплины 9
7 Фонд оценочных средств 9
7.1 Методы контроля и оценивания результатов обучения 9
7.2 Шкала и критерии оценивания результатов обучения 9
7.3 Оценочные средства 12

1 Цели, задачи и планируемые результаты обучения по дисциплине

Целями преподавания дисциплины являются:

• формирование компетенций, направленных на создание у студентов целостного представления об информации, методах ее получения, хранения, обработки и передачи; о роли информатики и месте информатики в современном обществе; понимание информационных процессов и технологий обработки данных; изучение логических основ ЭВМ и основных принципов компьютерного моделирования; формирование систематических знаний в области теоретических основ информатики.

Основными задачами освоения дисциплины являются:

сформировать представление об информатике как науке и отрасли индустрии;

- изучить концептуальные модели информатики;
- изучить математические основы информатики;
- изучить лингвистические основы информатики;
- сформировать знания в области теории кодирования и передачи информации..

Планируемые результаты обучения должны быть соотнесены с установленными в ОПОП индикаторами достижения компетенций.

Обучение по дисциплине «Теоретические основы информатики» направлено на формирование у обучающихся следующих компетенций:

Код и наименование компетенций	Индикаторы достижения компетенции
ОПК-1. Способен применять естественнонаучные и общеинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности.	**Знать**: Методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности. **Уметь**: Применять естественнонаучные и общеинженерные знания. **Владеть**: Методами математического анализа и моделирования, теоретического и экспериментального исследования.

2 Место дисциплины в структуре образовательной программы

Дисциплина относится к обязательной части (части, формируемой участниками образовательных отношений) блока Б1 «Дисциплины (модули)».

Дисциплина «Теоретические основы информатики» взаимосвязана логически и содержательно-методически со следующими дисциплинами (Интернет технологии, электронный бизнес и др.), а также рядом специальных дисциплин.

3 Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 3 зачетных(е) единиц(ы) (108 часов).

3.1 Виды учебной работы и трудоемкость (по формам обучения)

3.1.1 Заочная форма обучения

№ п/п	Вид учебной работы	Количество часов	Семестры	
			1	
1	**Аудиторные занятия**	**48**		
	В том числе:			
1.1	Лекции	16	16	
1.2	Семинарские/практические занятия			
1.3	Лабораторные занятия	32	32	
2	**Самостоятельная работа**	**60**		
	В том числе:			
2.1	Подготовка к лабораторным работам	60	60	
3	**Промежуточная аттестация**			
	Зачет/диф.зачет/экзамен		экзамен	
	Итого:	**108**	108	

3.2 Тематический план изучения дисциплины (по формам обучения)

3.2.1 Заочная форма обучения

№ п/п	Разделы/темы дисциплины	Трудоемкость, час					
		Всего	Аудиторная работа				Самостоятельная работа
			Лекции	Семинарские/ практические занятия	Лабораторные занятия	Практическая подготовка	
1	Раздел 1.						
1.1	Тема 1. Язык JAVA, классы, объекты, методы, переменные и типы данных	6	2				4
1.2	Тема 2. Операции в JAVA	6	2				4
1.3	Тема 3. Операторы	6	2				4
1.4	Тема 4. Массивы и строки	5	2				3
1.5	Тема 5. Классы и объекты, наследование	5	2				3
1.6	Тема 6. Абстрактные классы и интерфейсы	5	2				3
1.7	Тема 7. Работа с потоками ввода-вывода.	5	2				3
1.8	Тема 8. Обобщенное программирование	5	2				3

2	Раздел 2.						
2.1	Лабораторная работа № 1. Простое приложение в среде NetBeansЗадание	6			3		3
2.2	Лабораторная работа № 2. Создание классов и объектов в Java	6			3		3
2.3	Лабораторная работа № 3. Наследование классов	6			3		3
2.4	Лабораторная работа № 4. Абстрактные классы	6			3		3
2.5	Лабораторная работа № 5. Интерфейсы.	6			3		3
2.6	Лабораторная работа № 6 Работа с формулами	6			3		3
2.7	Лабораторная работа № 7 Разветвляющиеся программы	6			3		3
2.8	Лабораторная работа № 8 Одномерные массивы	6			3		3
2.9	Лабораторная работа № 9 Программирование исключительных ситуаций и событий языка программирования Java.	6			3		3
2.10	Лабораторная работа № 10 Работа с классами	6			3		3
2.11	Лабораторная работа № 11 Работа с текстом.	5			2		3
Итого		**108**	**16**		**32**		**60**

3.3 Содержание дисциплины

Раздел 1.

Тема 1. Язык JAVA, классы, объекты, методы, переменные и типы данных

О языке JAVA. Подготовка к работе с Java. Соглашение об именах. Классы, объекты и методы. Выполнение программы. Метод main. Структура проекта Java. Типы данных.

Тема 2. Операции в JAVA

Арифметические операции. Приоритет арифметических операций. Ассоциативность операций. Операции с числами с плавающей точкой. Логические операции. Операции сдвига.

Тема 3. Операторы

Основные операторы. Логические операторы. Битовые операторы. Операторы сравнения. Тернарный оператор. Оператор выбора switch. Оператор цикла do… while.

Тема 4. Массивы и строки

Одномерные массивы. Инициализация одномерного массива. Специальная форма оператора for. Присваивание массивов. Двумерные массивы. Инициализация двумерного массива.

Тема 5. Классы и объекты, наследование

Описание класса и создание объектов. Пример класса, содержащего только методы. Геттеры и сеттеры. Конструктор класса. Статические поля и методы. Наследование.

Тема 6. Абстрактные классы и интерфейсы

Абстрактные классы. Интерфейсы. Интерфейсные переменные. Наследование интерфейсов. Совмещение наследования и реализации.

Тема 7. Работа с потоками ввода-вывода.

Работа с потоками ввода-вывода. Иерархия потоков в Java. Класс OutputSrteam. Специализированные потоки. Преобразующие потоки. Стандартные потоки ввода-вывода. Файловые каналы.

Тема 8. Обобщенное программирование

Обобщенное программирование. Описание типов с шаблонами. Описание методов с шаблонами. Формальные параметры типа. Шаблоны с групповой подстановкой.

3.4 Тематика семинарских/практических и лабораторных занятий

3.4.1 Семинарские/практические занятия

Не предусмотрено.

3.4.2 Лабораторные занятия

Лабораторная работа № 1. Простое приложение в среде NetBeansЗадание

Лабораторная работа № 2. Создание классов и объектов в Java

Лабораторная работа № 3. Наследование классов

Лабораторная работа № 4. Абстрактные классы

Лабораторная работа № 5. Интерфейсы.

Лабораторная работа № 6 Работа с формулами

Лабораторная работа № 7 Разветвляющиеся программы

Лабораторная работа № 8 Одномерные массивы

Лабораторная работа № 9 Программирование исключительных ситуаций и событий языка программирования Java.

Лабораторная работа № 10 Работа с классами

Лабораторная работа № 11 Работа с текстом..

3.5 Тематика курсовых проектов (курсовых работ)

Не предусмотрено

4 Учебно-методическое и информационное обеспечение

4.1 Нормативные документы и ГОСТы

1. Федеральный государственный образовательный стандарт высшего образования (уровень бакалавриата) по направлению подготовки 09.03.03 «Прикладная информатика», утвержденный приказом Министерства образования и науки Российской Федерации от 19.09.2017 №922 «Об утверждении федерального государственного стандарта высшего образования – бакалавриат по направлению подготовки 09.03.03 Прикладная информатика. https://fgos.ru/fgos/fgos-09-03-03-prikladnaya-informatika-922/

4.2 Основная литература

1. Пономарчук, Ю. В. Программирование на языке Java : учебное пособие / Ю. В. Пономарчук, И. В. Кузнецов. — Хабаровск : ДВГУПС, 2021. — 103 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/259451 (дата обращения: 13.10.2023). — Режим доступа: для авториз. пользователей.

2. Программирование на java : учебно-методическое пособие / составитель Ю. А. Крыжановская. — Воронеж : ВГУ, 2012. — 52 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/357971 (дата обращения: 13.10.2023). — Режим доступа: для авториз. пользователей.

4.3 Дополнительная литература

1. Хабитуев, Б. В. Программирование на языке Java: практикум : учебное пособие / Б. В. Хабитуев. — Улан-Удэ : БГУ, 2020. — 94 с. — ISBN 978-5-9793-1548-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/171791 (дата обращения: 13.10.2023). — Режим доступа: для авториз. пользователей.

2. Болбот, О. М. Классы в языке программирования Java : учебно-методическое пособие / О. М. Болбот, В. В. Сидорик ; под редакцией В. В. Сидорика. — Минск : БНТУ, 2020. — 76 с. — ISBN 978-985-550-895-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/248009 (дата обращения: 13.10.2023). — Режим доступа: для авториз. пользователей.

4.4 Электронные образовательные ресурсы

1. https://online.mospolytech.ru/course/view.php?id=8596&notifyeditingon=1 - Теоретические основы информатики

4.5 Лицензионное и свободно распространяемое программное обеспечение

1. Операционная система, Windows 11 (или ниже) - Microsoft Open License
2. Офисные приложения, Microsoft Office 2013(или ниже) - Microsoft Open License
3. Netbeans

4.6 Современные профессиональные базы данных и информационные справочные системы

1.не предусмотрено

5 Материально-техническое обеспечение

5.1 Требования к оборудованию и помещению для занятий

Лабораторные работы и самостоятельная работа студентов должны проводиться в специализированной аудитории, оснащенной современной оргтехникой и персональными компьютерами с программным обеспечением в соответствии с тематикой изучаемого материала. Число рабочих мест в аудитории должно быть достаточным для обеспечения индивидуальной работы студентов. Рабочее место преподавателя должно быть оснащены современным компьютером с подключенным к нему проектором на настенный экран, или иным аналогичным по функциональному назначению оборудованием.

5.2 Требования к программному обеспечению

Для выполнения лабораторных работ и самостоятельной работы необходимо следующее программное обеспечение:

Microsoft Windows.

Веб-браузер, Chrome.

ПО, предоставленное преподавателем.

6 Методические рекомендации

6.1 Методические рекомендации для преподавателя по организации обучения

1. При подготовке к занятиям следует предварительно проработать материал занятия, предусмотрев его подачу точно в отведенное для этого время занятия. Следует подготовить необходимые материалы – теоретические сведения, задачи и др. При проведении занятия следует контролировать подачу материала и решение заданий с учетом учебного времени, отведенного для занятия.

2. При проверке работ и отчетов следует учитывать не только правильность выполнения заданий, но и оптимальность выбранных методов решения, правильность выполнения всех его шагов.

6.2 Методические указания для обучающихся по освоению дисциплины

Изучение дисциплины осуществляется в строгом соответствии с целевой установкой в тесной взаимосвязи учебным планом. Основой теоретической подготовки студентов являются аудиторные занятия, семинары и практики.

В процессе самостоятельной работы студенты закрепляют и углубляют знания, полученные во время аудиторных занятий, дорабатывают конспекты и записи, готовятся к промежуточной аттестации, а также самостоятельно изучают отдельные темы учебной программы.

На занятиях студентов, в том числе предполагающих практическую деятельность, осуществляется закрепление полученных, в том числе и в процессе самостоятельной работы, знаний. Особое внимание обращается на развитие умений и навыков установления связи положений теории с профессиональной деятельностью будущего специалиста.

Самостоятельная работа осуществляется индивидуально. Контроль самостоятельной работы организуется в двух формах:

самоконтроль и самооценка студента;

контроль со стороны преподавателей (текущий и промежуточный).

Текущий контроль осуществляется на аудиторных занятиях, промежуточный контроль осуществляется на зачете в письменной (устной) форме.

Критериями оценки результатов самостоятельной работы студента являются:

уровень освоения студентом учебного материала;

умения студента использовать теоретические знания при выполнении практических задач;

сформированность компетенций;

оформление материала в соответствии с требованиями..

7 Фонд оценочных средств

7.1 Методы контроля и оценивания результатов обучения

В процессе обучения используются следующие оценочные формы самостоятельной работы студентов, оценочные средства текущего контроля успеваемости и промежуточных аттестаций:

Лабораторные работы, экзамен.

7.2 Шкала и критерии оценивания результатов обучения

Показателем оценивания компетенций на различных этапах их формирования является достижение обучающимися планируемых результатов обучения по дисциплине.

Показатель	Критерии оценивания			
	2	3	4	5
ОПК-1. Способен применять естественнонаучные и общеинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности.				
ОПК-1.1. Знать: Методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности. **ОПК-1.2. Уметь:** Применять естественнонаучные и общеинженерные знания. **ОПК-1.3. Владеть:** Методами математического анализа и моделирования, теоретического и экспериментального исследования.	Обучающийся демонстрирует полное отсутствие или недостаточное соответствие материалу дисциплины знаний, указанных в индикаторах компетенций дисциплины «Знать» (см. п. 3).	Обучающийся демонстрирует неполное соответствие следующих знаний, указанных в индикаторах компетенций дисциплины «Знать» (см. п. 3). Допускаются значительные ошибки, проявляется недостаточность знаний, по ряду показателей, обучающийся испытывает значительные затруднения при оперировании знаниями при их переносе на новые ситуации.	Обучающийся демонстрирует частичное соответствие следующих знаний, указанных в индикаторах компетенций дисциплины «Знать» (см. п. 3). Но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях.	Обучающийся демонстрирует полное соответствие следующих знаний, указанных в индикаторах компетенций дисциплины «Знать» (см. п. 3). Свободно оперирует приобретенными знаниями.

Шкала оценивания результатов промежуточной аттестации и её описание:

Форма промежуточной аттестации: экзамен.

Промежуточная аттестация обучающихся в форме экзамена проводится по результатам выполнения всех видов учебной работы, предусмотренных учебным планом по данной дисциплине, при этом учитываются результаты текущего контроля успеваемости в течение семестра. Оценка степени достижения обучающимися планируемых результатов обучения по дисциплине проводится преподавателем, ведущим занятия по дисциплине методом экспертной оценки. По итогам промежуточной аттестации по дисциплине

выставляется оценка «отлично», «хорошо», «удовлетворительно» или «неудовлетворительно».

Шкала оценивания	**Описание**
Отлично	Выполнены все **обязательные условия подготовки студента к промежуточной аттестации**, предусмотренные программой дисциплины. Студент демонстрирует соответствие знаний, умений, навыков показателям, приведенным в таблицах, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены незначительные ошибки, неточности, затруднения при переносе знаний и умений на новые, нестандартные задачи.
Хорошо	Выполнены все **обязательные условия подготовки студента к промежуточной аттестации**, предусмотренные программой дисциплины. Студент демонстрирует соответствие знаний, умений, навыков показателям, приведенным в таблицах, оперирует приобретенными знаниями, умениями, навыками. В то же время при ответе допускает несущественные погрешности, задачи решает с недочетами, не влияющими на общий ход решения.
Удовлетворительно	Выполнены все **обязательные условия подготовки студента к промежуточной аттестации**, предусмотренные программой дисциплины. Студент демонстрирует соответствие знаний, умений, навыков показателям, приведенным в таблицах, оперирует приобретенными знаниями, умениями, навыками. Но показывает неглубокие знания, при ответе не допускает грубых ошибок или противоречий, однако в формулировании ответа отсутствует должная связь между анализом, аргументацией и выводами, в решении задач могут содержаться грубые ошибки. Для получения правильного ответа требуются уточняющие вопросы.
Неудовлетворительно	Не выполнены обязательные **условия подготовки студента к промежуточной аттестации**, предусмотренные программой дисциплины, ИЛИ студент демонстрирует неполное соответствие знаний, умений, навыков приведенным в таблицах, допускаются значительные ошибки, проявляется отсутствие знаний, умений, навыков по ряду показателей, студент испытывает значительные затруднения при оперировании знаниями и умениями.

7.3 Оценочные средства

7.3.1 Промежуточная аттестация

Перечень экзаменационных вопросов по дисциплине «Теоретические основы информатики»

1. Структура проекта Java. Комментарии в Java.
2. Переменные. Ключевое слово VAR.
3. Примитивные типы данных. Приведение типов (Явное и Неявное).
4. Консольный ввод/вывод информации в Java.
5. Преобразование типов данных. Явное и неявное преобразование. Потеря данных при преобразовании.
6. Арифметические операции в Java. Приоритет арифметических операций и ассоциативность в Java.
7. Операции с числами с плавающей точкой. Побитовые операции.
8. Логические операции в Java.
9. Операции сдвига в Java.
10. Условные выражения. Операции сравнения. Логические операции.
11. Операции присваивания. Приоритет операций присваивания.
12. Четыре основные группы операторов в Java. Арифметические операторы.
13. Четыре основные группы операторов в Java. Логические операторы.
14. Четыре основные группы операторов в Java. Битовые операторы.
15. Четыре основные группы операторов в Java. Операторы сравнения.
16. Тернарный оператор в Java.
17. Управляющие инструкции. Условный оператор If.
18. Управляющие инструкции. Оператор выбора Switch.
19. Операторы цикла. Оператор цикла while.
20. Операторы цикла. Оператор цикла do… while.
21. Операторы цикла. Оператор цикла for.
22. Операторы досрочного выхода break, continue и return.
23. Массив. Объявление и индексация массива.
24. Одномерный массив. Инициализация одномерного массива.
25. Присваивание массивов в Java.
26. Двумерный массив. Инициализация двумерного массива.
27. Методы для операций с массивами equals(),copyOfRange().
28. Методы для операций с массивами toString(),sort().
29. Методы для операций с массивами binarySearch().
30. Строки. Методы для операций со строками charAt(), contains().
31. Строки. Методы для операций со строками endsWith(), startsWith(), equals().
32. Строки. Методы для операций со строками equalsIgnoreCase(), length(), split().
33. Строки. Методы для операций со строками substring(), toUpperCase() /toLowerCase(), trim().
34. Класс. Описание класса и создание объектов в Java.
35. Класс. Геттеры и сеттеры в Java.
36. Класс. Конструктор класса. Пример использования конструкторов.
37. Класс. Статические поля и методы.
38. Класс. Наследование классов.
39. Абстрактные классы. Пример реализации абстрактного класса.
40. Интерфейс. Интерфейсные переменные. Пример реализации интерфейса.

41. Интерфейс. Методы в интерфейсе. Наследование интерфейсов.
42. Работа с потоками ввода-вывода данных. Иерархия потоков в Java.
43. Класс java.lang.System. Стандартные потоки ввода-вывода данных.
44. Класс java.nio.Buffer. Работа с буферами памяти.
45. Каналы передачи данных в Java. Файловые и Сетевые каналы.
46. Использование легковесных процессов в Java. Поток и реализация потока.
47. Поток. Распределение приоритета между потоками. Класс java.lang.ThreadGroup.
48. Поток. Взаимодействие и синхронизация потоков. Модификатор volatile.
49. Обобщенное программирование. Шаблон. Принцип использования шаблонов.
50. Обобщенное программирование. Шаблон. Описание типов с шаблонами. Формальные параметры типа.
51. Обобщенное программирование. Шаблон. Описание методов с шаблонами
52. Обобщенное программирование. Шаблон. Шаблоны с групповой подстановкой.
53. Сериализация объектов. Интерфейс java.io.Serializable.
54. Сериализация объектов. Класс java.io.ObjectOutputStream.
55. Сериализация объектов. Класс java.io.ObjectInputStream.
56. Сериализация объектов. Интерфейс java.io.Externalizable. Контроль версий сериализуемого класса.
57. Регулярные выражения в Java. Matcher.
58. Регулярные выражения в Java. replaceFirst и replaceAll.
59. Регулярные выражения в Java. Группы сбора.
60. Регулярные выражения в Java. Метасимволы.

Задачи к экзамену по дисциплине «Теоретические основы информатики».

1. Создать приложение, возводящее целое положительной число X в целую положительную степень Y. Результат вывести на экран.
2. Создать приложение, находящее сумму цифр двух положительных и трех отрицательных целых чисел. Ограничить ввод числа тремя знаками. Результат вывести на экран.
3. Создать приложение, находящее сумму цифр положительного целого числа. Ограничить ввод числа шестью знаками. Результат вывести на экран.
4. Создать приложение, находящее максимальный элемент двумерного целочисленного массива A. Результат вывести на экран.
5. Подготовьте фрагмент программы, в котором должны вводиться 10 чисел. Если будет введено число 0, ввод должен прекратиться.
6. Заполнить массив из десяти элементов значениями, вводимыми с клавиатуры в ходе выполнения программы. Результат вывести на экран.
7. Заполнить двумерный массив из трех элементов значениями, вводимыми с клавиатуры в ходе выполнения программы. Результат вывести на экран.
8. Создать консольное приложение для вычисления арифметического выражения с 3 переменными, каждая переменная должна запрашиваться у пользователя отдельно.
9. Число π примерно равно 3,1415926. Вывести на экран это число с тремя цифрами в дробной части. Текст '3.142' не использовать.
10. Составить программу вывода на экран числа, вводимого с клавиатуры. Выводимому числу должно предшествовать сообщение «Вы ввели число».

11. Составить программу вывода на экран числа, вводимого с клавиатуры. После выводимого числа должно следовать сообщение «– вот какое число Вы ввели»
12. Составить программу вывода на экран любого элемента массива по его индексу.
13. Составить программу поиска любого элемента массива по его значению. Результат вывести на экран.
14. Составить программу вывода на экран в одну строку трех любых чисел, вводимых с клавиатуры, с двумя пробелами между ними.
15. Вывести на экран числа 5, 10 и 21 одно под другим.
16. Возраст Тани – X лет, а возраст Саши – Y лет. Найти их средний возраст, а также определить, на сколько отличается возраст каждого ребенка от среднего значения. Результат вывести на экран.
17. Даны два числа. Если квадратный корень из второго числа меньше первого числа, то увеличить второе число в пять раз. Результат вывести на экран.
18. Даны шесть целых чисел, 2 из них отрицательные. Числа вводятся с клавиатуры. Определить сумму положительных из них. Результат вывести на экран.
19. Напечатать «столбиком» все четные числа от 20 до 35.
20. Напечатать «столбиком» все нечетные числа от 20 до 35.
21. Напечатать «столбиком» квадраты всех целых чисел от a до 50 (значение a вводится с клавиатуры; $a \leq 50$).
22. Напечатать ряд чисел 20 в виде: 20 20 20 20 20 20 20 20 20 20.
23. Одноклеточная амеба каждые 3 часа делится на 2 клетки. Определить, сколько клеток будет через 3, 6, 9, ..., 24 часа, если первоначально была одна амеба. Результат вывести на экран.
24. Гражданин 1 марта открыл счет в банке, вложив 1000 руб. Через каждый месяц размер вклада увеличивается на 2 % от имеющейся суммы. Определить прирост суммы вклада за первый, второй, ..., пятый месяц; Результат вывести на экран.
25. Даны натуральные числа a и b ($a > b$). Определить результат целочисленного деления a на b, используя стандартную операцию целочисленного деления;
26. Разработать фрагмент программы, в котором пользователь должен ввести четное число. В случае ввода нечетного числа на экран должно выводиться сообщение об ошибке, после чего действия должны повторяться до ввода правильного значения.
27. Среди чисел 1, 4, 9, 16, 25, ... найти первое число, большее n. Задачу решить с использованием оператора цикла с условием. Число n задается с клавиатуры.
28. Среди чисел 1, 4, 9, 16, 25, ... найти первое число, большее n. Задачу решить без использования оператора цикла с условием. Число n задается с клавиатуры.
29. Дан массив целых чисел. Вывести на экран сначала его четные элементы, затем нечетные.
30. Дан массив целых отрицательных и положительных чисел. Вывести на экран сначала его отрицательные четные элементы, затем положительные нечетные элементы.