

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
СТАВРОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ**

**УТВЕРЖДАЮ**

Директор/Декан  
института экономики, финансов и  
управления в АПК  
Гунько Юлия Александровна

\_\_\_\_\_  
\_\_\_\_\_  
«\_\_» \_\_\_\_\_ 20\_\_ г.

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ (ОЦЕНОЧНЫХ МАТЕРИАЛОВ)**

**Б1.О.19 Технологии программирования**

38.03.05 Бизнес-информатика

Электронный бизнес

бакалавр

очная

# 1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Процесс изучения дисциплины направлен на формирование следующих компетенций ОП ВО и овладение следующими результатами обучения по дисциплине:

Код и наименование компетенции	Код и наименование индикатора достижения	Перечень планируемых результатов обучения по дисциплине
<p>ОПК-3 Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации;</p>	<p>ОПК-3.2 Разрабатывает алгоритмы и программы для практического применения в сфере ИКТ</p>	<p><b>знает</b> алгоритмы и программы для практического применения в сфере ИКТ</p>
		<p><b>умеет</b> разрабатывать алгоритмы и программы для практического применения в сфере ИКТ</p>
		<p><b>владеет навыками</b> навыками разработки алгоритмов и программ для практического применения в сфере ИКТ</p>
<p>ОПК-6 Способен выполнять отдельные задачи в рамках коллективной научно-исследовательской, проектной и учебно-профессиональной деятельности для поиска, выработки и применения новых решений в области информационно-коммуникационных технологий.</p>	<p>ОПК-6.4 Решает профессиональные задачи с применением естественнонаучных и инженерных знаний, методов математического анализа и моделирования</p>	<p><b>знает</b> естественнонаучные и инженерные знания, методы математического анализа и моделирования</p>
		<p><b>умеет</b> Решать профессиональные задачи с применением естественнонаучных и инженерных знаний, методов математического анализа и моделирования Знать</p>
		<p><b>владеет навыками</b> навыками решения профессиональных задач с применением естественнонаучных и инженерных знаний, методов математического анализа и моделирования Знать</p>
<p>ПК-1 Управление информационными ресурсами</p>	<p>ПК-1.2 Управление информацией из различных источников</p>	<p><b>знает</b> принципы работы с источниками данных, методы их верификации и систематизации информации</p>
		<p><b>умеет</b> осуществлять сбор, фильтрацию и структурирование информации из разнородных источников</p>
		<p><b>владеет навыками</b> технологиями анализа данных, инструментами мониторинга и управления информационными потоками</p>

## 2. Перечень оценочных средств по дисциплине

№	Наименование раздела/темы	Семестр	Код индикаторов достижения компетенций	Оценочное средство проверки результатов достижения индикаторов компетенций
1.	1 раздел. 1			
1.1.	Введение в алгоритмизацию и программирование	2	ОПК-3.2, ОПК-6.4, ПК-1.2	Тест
1.2.	Структурный подход к программированию	2	ОПК-3.2, ОПК-6.4, ПК-1.2	Тест
1.3.	Кт 1	2	ОПК-3.2, ОПК-6.4, ПК-1.2	Тест
1.4.	Модульное программирование. Программирование абстрактных типов данных	2	ОПК-3.2, ОПК-6.4, ПК-1.2	Тест
1.5.	Объектно-ориентированное программирование	2	ОПК-3.2, ОПК-6.4, ПК-1.2	Тест
1.6.	Кт 2	2	ОПК-3.2, ОПК-6.4, ПК-1.2	Тест
	Промежуточная аттестация			Эк

## 3. Оценочные средства (оценочные материалы)

Примерный перечень оценочных средств для текущего контроля успеваемости и промежуточной аттестации

№ п/п	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в фонде (Оценочные материалы)
<b>Текущий контроль</b>			
<b>Для оценки знаний</b>			
1	Тест	Система стандартизированных заданий, позволяющая автоматизировать процедуру измерения уровня знаний и умений обучающегося.	Фонд тестовых заданий
<b>Для оценки умений</b>			
<b>Для оценки навыков</b>			
<b>Промежуточная аттестация</b>			

2	Экзамен	Средство контроля усвоения учебного материала и формирования компетенций, организованное в виде беседы по билетам с целью проверки степени и качества усвоения изучаемого материала, определить необходимость введения изменений в содержание и методы обучения.	Комплект экзаменационных билетов
---	---------	--	----------------------------------

**4. Примерный фонд оценочных средств для проведения текущего контроля и промежуточной аттестации обучающихся по дисциплине (модулю) "Технологии программирования"**

*Примерные оценочные материалы для текущего контроля успеваемости*

ЗНАТЬ

Задание 1

Стартап разрабатывает новый мобильный банковский продукт в условиях высокой неопределённости требований и жёстких сроков выхода на рынок. Какой подход к управлению проектом будет наиболее предпочтителен?

Варианты:

- A) Каскадная модель (Waterfall)
- B) Гибкая методология (Agile, например, Scrum)
- C) V-образная модель
- D) RUP (Rational Unified Process)

Ответ: B

Задание 2

В бизнес-логике расчёта скидок для интернет-магазина один и тот же алгоритм проверки статуса постоянного клиента используется в 12 разных местах кода. Что это нарушает и как лучше исправить?

Варианты:

- A) Это нарушение принципа DRY. Нужно вынести алгоритм в отдельную функцию/метод и вызывать её.
- B) Это нормальная практика для ускорения работы программы.
- C) Это нарушение принципа KISS. Нужно удалить лишние проверки.
- D) Ничего исправлять не нужно, если код работает.

Ответ: A

Задание 3

Команда из 5 человек одновременно работает над разными функциями веб-приложения для CRM-системы. Какой инструмент критически необходим для слияния их изменений, отслеживания истории правок и отката ошибок?

Варианты:

- A) Система управления базами данных (СУБД)
- B) Интегрированная среда разработки (IDE)
- C) Система контроля версий (Git)
- D) Система баг-трекинга (Jira)

Ответ: C

Задание 4

После завершения разработки модуля «Формирование платёжной ведомости» необходимо проверить, корректно ли он взаимодействует с модулем «Налоговый учёт». Какой тип тестирования описывается?

Варианты:

- A) Модульное (Unit) тестирование
- B) Интеграционное (Integration) тестирование

- C) Приёмочное (UAT) тестирование
- D) Нагрузочное (Load) тестирование

Ответ: B

Задание 5

На каком этапе жизненного цикла программного обеспечения определяются ключевые бизнес-требования, ограничения бюджета и сроки?

Варианты:

- A) Проектирование (Design)
- B) Реализация (Implementation)
- C) Сопровождение (Maintenance)
- D) Сбор  требований (Requirements Gathering & Analysis)

Ответ: D

Задание 6

В системе управления складом есть различные алгоритмы расчёта стоимости доставки (для курьера, почты, самовывоза). Требуется организовать код так, чтобы можно было легко добавлять новые алгоритмы, не изменяя основной код заказа. Какой паттерн проектирования лучше применить?

Варианты:

- A) Фабричный метод (Factory Method)
- B) Одиночка (Singleton)
- C) Стратегия (Strategy)
- D) Наблюдатель (Observer)

Ответ: C

Задание 7

Чтобы успеть к запуску промо-акции, команда разработки внедрила функцию быстрой оплаты, используя упрощённые, но неоптимальные решения. Что образуется в коде и проекте вследствие такого решения?

Варианты:

- A) Накопленный капитал
- B) Технический долг
- C) Бизнес-актив
- D) Резервная копия

Ответ: B

Задание 8

Для высоконагруженного сервиса аналитики в реальном времени, который обрабатывает поток кликов с сайта, требуется хранить слабоструктурированные данные с возможностью горизонтального масштабирования. Какой тип базы данных наиболее предпочтителен?

Варианты:

- A) Реляционная СУБД (MySQL, PostgreSQL)
- B) Документоориентированная NoSQL БД (MongoDB)
- C) Графовая БД (Neo4j)
- D) SQLite

Ответ: B

Задание 9

Что является основной целью рефакторинга кода?

Варианты:

- A) Добавление новой функциональности для пользователя.
- B) Исправление критических ошибок (багов).
- C) Улучшение внутренней структуры кода без изменения его внешнего поведения.
- D) Ускорение работы программы за счёт оптимизации алгоритмов.

Ответ: C

Задание 10

Как называется практика и набор инструментов, которые позволяют автоматически собирать, тестировать и разворачивать приложение при каждом изменении кода, что ускоряет доставку обновлений пользователям?

Варианты:

- A) Система управления проектами (MS Project)

- В) Интеграционная среда разработки (Visual Studio)
- С) Непрерывная интеграция и непрерывная поставка (CI/CD)
- Д) Виртуализация (Docker)

Ответ: С

Задание 11

Кто в команде разработки преимущественно отвечает за перевод требований бизнеса (заказчика) в формализованные спецификации и пользовательские сценарии (User Stories), понятные разработчикам?

Варианты:

- А) Системный администратор
- В) Тестировщик (QA Engineer)
- С) Бизнес-аналитик
- Д) Тимлид (Team Lead)

Ответ: С

Задание 12

С точки зрения бизнеса, какое из перечисленных является главным экономическим преимуществом использования облачной инфраструктуры (IaaS, например, AWS, Azure) вместо содержания собственного серверного оборудования?

Варианты:

- А) Перевод капитальных затрат (CAPEX) в операционные (OPEX).
- В) Полный контроль над физическим оборудованием.
- С) Гарантированная максимальная скорость работы.
- Д) Отсутствие необходимости в IT-специалистах.

Ответ: А

Задание 13

Чем микросервисная архитектура принципиально отличается от монолитной?

Варианты:

- А) Приложение разбито на множество небольших, слабосвязанных сервисов, развёртываемых независимо.
- В) Приложение написано на более современных языках программирования.
- С) Все компоненты приложения тесно связаны и развёртываются как единое целое.
- Д) Она всегда работает быстрее монолита.

Ответ: А

Задание 14

Злоумышленник смог внедрить в поле ввода на сайте фрагмент SQL-кода, который изменил запрос к базе данных и позволил получить доступ к чужим данным. Какая уязвимость была использована?

Варианты:

- А) XSS (Межсайтовый скриптинг)
- В) SQL-инъекция
- С) DDoS-атака
- Д) Фишинг

Ответ: В

Задание 15

В User Story для функции «Восстановление пароля» указано: «Пользователь должен получить временную ссылку для сброса пароля на email». Что из перечисленного является хорошим, проверяемым критерием приемлемости?

Варианты:

- А) Ссылка должна быть красивой.
- В) При вводе зарегистрированного email в форму восстановления пароля система отправляет письмо с уникальной ссылкой, активной в течение 1 часа, на этот адрес.
- С) Система должна быстро отправлять письма.
- Д) Это важная функция для безопасности.

Ответ: В

Задание 1

Этапы классического жизненного цикла разработки ПО (Waterfall). Установите правильный порядок этапов.

1. Сопровождение
2. Проектирование
3. Тестирование
4. Анализ требований
5. Реализация (Кодирование)

Ответ: 4 → 2 → 5 → 3 → 1

Задание 2

Приоритет действий при получении задачи на исправление бага (bugfix). Расположите действия в логическом порядке.

1. Написать автоматический тест, воспроизводящий баг.
2. Проанализировать логи и стэк-трейс ошибки.
3. Воспроизвести баг на тестовом окружении.
4. Отправить исправление в репозиторий и запустить сборку.
5. Локализовать причину в исходном коде.

Ответ: 2 → 3 → 1 → 5 → 4

Задание 3

Стадии процесса рефакторинга кода. Установите последовательность.

1. Внести минимальные изменения в код.
2. Убедиться, что существующие тесты проходят.
3. Определить «запах кода» (code smell), требующий улучшения.
4. Написать тесты для покрытия рефакторимого участка (если их нет).
5. Провести рефакторинг, не меняя поведения.

Ответ: 3 → 4 → 2 → 5 → 1

Задание 4

Установите порядок создания простой пользовательской истории в формате INVEST

1. Уточняющий критерий приемки (Acceptance Criteria)
2. Роль (As a...)
3. Цель/ценность (So that...)
4. Действие (I want to...)

Ответ: 2 → 4 → 3 → 1

Задание 5

Установите правильный порядок типичного workflow.

1. Создать новую ветку (git checkout -b feature\_branch)
2. Зафиксировать изменения (git commit -m "...")
3. Влить ветку в main (git merge / pull request)
4. Сделать коммиты локально
5. Получить последние изменения с главной ветки (git pull origin main)

Ответ: 5 → 1 → 4 → 2 → 3

Задание 6

Установите соответствие между артефактами гибкой методологии (Scrum) и их описанием  
Артефакты:

- А. Бэклог продукта (Product Backlog)
- В. Бэклог спринта (Sprint Backlog)
- С. Инкремент продукта (Increment)

Описания:

1. Упорядоченный список всего, что может понадобиться в продукте, источник требований.
2. Рабочая версия продукта с новыми функциями, выполненных в конце спринта.
3. Набор элементов из бэклога продукта, выбранных для спринта, план реализации.

Ответ: А-1, В-3, С-2

Задание 7

Установите соответствие типов тестирования и их целей

Типы тестирования:

- А. Модульное (Unit)

В. Интеграционное (Integration)

С. Приемочное (Acceptance)

Цели:

1. Проверить соответствие системы бизнес-требованиям и готовность к использованию заказчиком.
2. Проверить работу отдельного класса, функции или модуля в изоляции.
3. Проверить взаимодействие между несколькими модулями, компонентами или системами.

Ответ: А-2, В-3, С-1

Задание 8

Установите соответствие «запахов кода» (code smells) и возможных методов рефакторинга

Запахи кода:

А. Длинный метод (Long Method)

В. Повторение кода (Duplicate Code)

С. Большой класс (Large Class)

Методы рефакторинга:

1. Выделение метода (Extract Method), выделение класса (Extract Class)
2. Выделение метода (Extract Method), выделение подкласса (Extract Subclass)
3. Выделение метода (Extract Method), перенос метода (Move Method)

Ответ: А-1, В-3, С-2

Задание 9

Установите соответствие видов архитектурных шаблонов и их ключевых характеристик

Шаблоны:

А. Многослойная архитектура (Layered Architecture)

В. Микросервисная архитектура (Microservices)

С. Событийно-ориентированная архитектура (Event-Driven)

Характеристики:

1. Система состоит из небольших, независимо развертываемых сервисов.
2. Компоненты общаются через асинхронную отправку и обработку событий.
3. Система разделена на слои (представление, бизнес-логика, данные), каждый зависит от нижележащего.

Ответ: А-3, В-1, С-2

Задание 10

Установите соответствие принципов SOLID и их кратких трактовок

Принципы:

А. Single Responsibility

В. Open/Closed

С. Liskov Substitution

Трактовки:

1. Классы должны быть открыты для расширения, но закрыты для модификации.
2. Дочерний класс должен быть способен заменить родительский, не ломая программу.
3. У класса должна быть только одна причина для изменения.

Ответ: А-3, В-1, С-2

Задание 11

Установите соответствие видов NoSQL баз данных и типичных сценариев использования в бизнесе

Виды БД:

А. Документная (MongoDB)

В. Колоночная (Cassandra)

С. Графовая (Neo4j)

Сценарии:

1. Хранение иерархических каталогов товаров, профилей пользователей с динамическими атрибутами.
2. Аналитика больших данных, обработка событий с высокой скоростью записи.
3. Рекомендательные системы, анализ социальных связей, обнаружение мошенничества.

Ответ: А-1, В-2, С-3

## Задание 12

Установите соответствие ролей в команде разработки и их ключевых задач

Роли:

- A. Бизнес-аналитик (Business Analyst, BA)
- B. Разработчик (Developer)
- C. Владелец продукта (Product Owner, PO)

Задачи:

1. Формирует видение продукта, расставляет приоритеты в бэклоге, максимизирует ценность.
2. Выявляет потребности стейкхолдеров, формализует требования, пишет пользовательские истории.
3. Проектирует, пишет, тестирует и отлаживает код согласно техническому заданию.

Ответ: A-2, B-3, C-1

## Задание 13

Установите соответствие инструментов и их основного назначения в процессе разработки

Инструменты:

- A. Jira / YouTrack
- B. Jenkins / GitLab CI
- C. Swagger / OpenAPI

Назначение:

1. Непрерывная интеграция и непрерывное развертывание (CI/CD).
2. Описание, создание и документирование REST API.
3. Отслеживание задач, багов и agile-планирование.

Ответ: A-3, B-1, C-2

## Задание 14

Установите соответствие метрик качества ПО и их описания

Метрики:

- A. Cyclomatic Complexity
- B. Code Coverage
- C. Mean Time To Recovery (MTTR)

Описания:

1. Мера того, какая часть исходного кода выполнена при прохождении тестов.
2. Среднее время, необходимое для восстановления работоспособности системы после сбоя.
3. Количество линейно независимых путей в коде, мера его сложности.

Ответ: A-3, B-1, C-2

## Задание 15

Установите соответствие паттернов проектирования (GoF) и решаемых проблем

Паттерны:

- A. Фасад (Facade)
- B. Стратегия (Strategy)
- C. Наблюдатель (Observer)

Проблемы:

1. Необходимость уведомлять множество объектов об изменении состояния другого объекта.
2. Предоставление простого интерфейса к сложной подсистеме.
3. Возможность выбирать алгоритм из семейства алгоритмов во время выполнения.

Ответ: A-2, B-3, C-1

## Задание 1

Этот принцип программирования (из аббревиатуры SOLID) гласит, что класс должен быть открыт для расширения, но закрыт для модификации. Напишите его название.

Ответ: Принцип открытости

## Задание 2

Как называется список всех функций, улучшений и исправлений, запланированных для продукта, который постоянно обновляется и приоритизируется? (Ключевой артефакт в Scrum).

Ответ: Бэклог продукта

### Задание 3

Какое минимальное число программистов должно быть в команде, чтобы практика парного программирования была возможна?

Ответ: 2

### Задание 4

Как называется процесс объединения копий рабочих версий (веток) из разных источников в единую ветку, например, в main? (Ключевое действие в Git).

Ответ: Слияние

### Задание 5

Если для метода `calculateDiscount(order)` в качестве входных данных мы передаем заказ с отрицательной суммой, чтобы проверить, как программа обрабатывает некорректные данные, — какой это вид тестирования?

Ответ: Негативное тестирование

### Задание 6

Какой шаблон проектирования позволяет объектам оповещать другие объекты об изменении своего состояния? (Часто используется в системах обработки событий).

Ответ: Наблюдатель

### Задание 7

Как называется язык разметки, который используется для описания и документирования REST API в удобной для человека и машины форме? (Пример инструмента — Swagger).

Ответ: OpenAPI

### Задание 8

В контексте жизненного цикла ПО, как называется этап, на котором команда исправляет найденные ошибки, адаптирует продукт к новому окружению и вносит небольшие улучшения?

Ответ: Сопровождение

### Задание 9

Если вы измеряете процент исходного кода, который выполняется вашими автоматическими тестами, как называется эта метрика качества?

Ответ: Покрытие кода

### Задание 10

Какой ключевой антипаттерн в управлении проектами описывает ситуацию, когда новая функция добавляется в проект просто потому, что она кажется интересной, а не потому, что она необходима? («Потому что можем»).

Ответ: Золотая молотка

### Задание 11

В модели CI/CD аббревиатура «CI» расшифровывается как...

Ответ: Непрерывная интеграция

### Задание 12

Как называется документ или его раздел, который содержит четкие и проверяемые условия, при которых пользовательская история или функция считается завершенной и работоспособной?

Ответ: Критерии приемки

### Задание 13

Если база данных хранит данные в виде узлов, связей между ними и их свойств, чтобы эффективно работать со сложными взаимосвязями (например, в соцсетях), то к какому основному типу NoSQL-БД она относится?

Ответ: Графовая

### Задание 14

Как называется практика, при которой репозиторий кода имеет одну главную ветку, а вся разработка ведется через создание короткоживущих веток для отдельных задач?

Ответ: Ветвление по функциональности

### Задание 15

Какой показатель (KPI) в DevOps измеряет среднее время, которое требуется команде на восстановление работоспособности системы после сбоя?

Ответ: MTTR

**Примерные оценочные материалы  
для проведения промежуточной аттестации (зачет, экзамен)  
по итогам освоения дисциплины (модуля)**

1. Системы программирования: основные понятия и определения
2. Состав системы программирования
3. Языки программирования: основные понятия и определения
4. Уровни языков программирования
5. Классификация языков программирования
6. Основные понятия кодирования информации и системы счисления
7. Виды систем счисления
8. Двоичная системы счисления
9. Десятичная системы счисления
10. Восьмеричная система счисления
11. Шестнадцатеричная система счисления
12. Порядок решения задачи алгоритмизации
13. Этапы решения задач на ПК
14. Алгоритм и его свойства
15. Способы описания алгоритмов
16. Словесно-формульный способ описания алгоритмов
17. Запись алгоритма на псевдокоде
18. Программный способ описания алгоритмов
19. Графический способ описания алгоритмов
20. Виды алгоритмов и принципы их составления
21. Характеристика линейных и разветвляющихся алгоритмов.
22. Алгоритм циклических вычислительных процессов, вложенные циклы.
23. Алгоритм комбинированных вычислительных процессов
24. Понятия и типы массивов
25. Алгоритмизация задач по обработке одномерных массивов
26. Алгоритмизация задач по обработке многомерных массивов
27. Язык программирования Python: область использования
28. Язык программирования Python: назначения
29. Язык программирования Python: основные проекты
30. История создания языка Python
31. Версии языка Python
32. Возможности языка Python и его предпосылки применения в обучении
33. Преимущества и недостатки Python
34. Установка и работа в Python
35. Синтаксис языка Python
36. Алфавит языка Python
37. Алфавит и словарь языка Python
38. Типы данных, используемых в Python
39. Выражения и операции в Python
40. Операторы присваивания, ввода, вывода, комментариев в Python
41. Стандартные функции в Python
42. Разработка на Python: плюсы и минусы
43. Примеры использования Python крупными компаниями
44. Достоинства и недостатки Python для разработки
45. Трудоустройство и средняя зарплата Python-разработчика
46. Должности и требования к Python-разработчику
47. Интегрированные среды для Python
48. Сравнение Python с другими языками программирования
49. Популярные языки программирования в 2024 году
50. Операторы ветвления в Python: формы, составные условия, вложенные ветвления
51. Операторы цикла в Python: цикл с предусловием (пример)
52. Операторы цикла в Python: цикл с постусловием (пример)

53. Операторы цикла в Python: цикл с параметром (пример)
54. Массивы в Python: понятие, виды, описание
55. Вывод массива на экран в Python
56. Способы заполнения массивов в Python
57. Сортировка массива в Python
58. Строковые величины в языке Python
59. Процедуры в языке Python
60. Функции в языке Python

Примерные практические задания на экзамене  
Язык программирования Python

1. Вычислить и вывести на экран монитора в виде таблицы значения функции, заданной с помощью ряда Тейлора, на интервале от  $X_{нач}$  до  $X_{кон}$  с шагом  $dx$  и точностью  $\epsilon$ . Таблицу снабдить заголовком и шапкой. Каждая строка таблицы должна содержать значение аргумента, значение функции и количество просуммированных членов ряда

2. Даны две матрицы одного порядка  $m * n$  ( $m$  строк  $\times$   $n$  столбцов).

Написать программу сложения, вычитания и транспонирования матриц.

1. Сложение и вычитание:

2. Транспонирование

3. Написать программу на Python, которая по введённому значению аргумента вычисляет значение функции, заданной в виде графика.

Гиперболический косинус может быть описан формулой:

4. Вывести на экран средствами пакета Tkinter в графическом режиме графики двух функций на интервале от  $X_{нач}$  до  $X_{кон}$  с шагом  $dx$ . Первая функция задана с помощью ряда Тейлора, ее вычисление должно выполняться с точностью  $\epsilon$ . Значение параметра  $b$  для второй функции вводится с клавиатуры. Графики должны быть плавными и различаться цветами.

5. Дана целочисленная прямоугольная матрица. Определить количество столбцов, не содержащих ни одного нулевого элемента. Переставляя строки заданной матрицы, расположить их в соответствии с ростом характеристик.

**Темы письменных работ (эссе, рефераты, курсовые работы и др.)**

Тематика рефератов

1. История языков программирования.
2. Язык компьютера и человека.
3. Объектно-ориентированное программирование.
4. Непроцедурные системы программирования.
5. Искусственный интеллект и логическое программирование.
6. Языки манипулирования данными в реляционных моделях.
7. Макропрограммирование в среде Microsoft OFFICE.
8. «Визуальное» программирование. VISUAL BASIC, C, PROLOG.
9. Все о DELPHI.
10. Программирование на HTML, JAVA.
11. Издательская система TeX как система программирования.
12. Современные парадигмы программирования. Что дальше?
13. Никлаус Вирт. Структурное программирование. Pascal и Modula.
14. Что мы знаем о Fortran?
15. История языка Бейсик.
16. Язык Ассемблера.
17. Алгоритмический язык Ершова.
18. Все о Logo-мирах.
19. История программирования в лицах.
20. Язык программирования ADA.
21. Язык программирования PL/1.
22. Язык программирования Algol.

23. Язык программирования Си.
24. О фирмах-разработчиках систем программирования.
25. Языки программирования в СУБД.
26. О системах программирования для учебных целей.

#### Тематика курсовой работы

1. Значение выбора языка, платформы и технологии программирования.
2. Важность структурирования программ и использования принципа модульности.
3. Варианты использования автокодов для оптимизации процессов.
4. Виды диаграмм и их значение в программировании.
5. Влияние программистов на эффективность организации рабочего процесса предприятий.
6. Значение процедурного программирования в 21 веке.
7. Конструкция императивных языков.
8. Макросы в ассемблере.
9. Оптимизация программного обеспечения для успешной и бесперебойной работы предприятия.
10. Особенности структурного программирования.
11. Проблемы современного программирования и их решение.
12. Программирование графических приложений с использованием графического интерфейса устройств (GDI+).
13. Особенности многоязыкового программирования.
14. Программирование как неотъемлемая часть развития общества.
15. Значение выбора языка, платформы и технологии программирования.
16. Программирование приложений с использованием библиотеки классов .NET Framework.
17. Программирование приложений, поддерживающих многодокументный интерфейс (MDI).
18. Программирование приложений, поддерживающих однодокументный интерфейс (SDI).
19. Программирование приложений, построенных на основе архитектуры «Документ/Представление»
20. Программирование с информационным сервером INTERNET (ISAPI).
21. Программирование сетевых приложений с использованием функций API.
22. Практическое использование генетического программирования.
23. Расширение областей программирования как следствие цифровизации современного общества.
24. Создание и индексация массивов.
25. Способы разработки мобильных приложений.
26. Сфера применения генерации случайных чисел.
27. Типизация в языках программирования.
28. Функции. Их параметры и аргументы в современном программировании.
29. Интерфейс во взаимодействии с другими системами программирования.
30. Язык СИ. Достоинства, недостатки и варианты применения.