
ТИПОВОЕ ЗАДАНИЕ ОЛИМПИАДЫ

1. Теоретическое задание (время выполнения ~25-30 минут)

1) Какому промежутку (отрезку) соответствует данное неравенство:

$$222_3 - x > 55_6$$

Варианты:

- a) $(-\infty ; 0)$
- b) $(0 ; +\infty)$
- c) $(4 ; +\infty)$
- d) $(-\infty ; 26)$

Ответ: _____

(время выполнения ~5 минут)

2) Для хранения растрового изображения используется палитра из 4 цветов. Известно, что изображение имеет размер 1024 на 256 пикселей. Каков минимально возможный информационный объем этого изображения в Килобайтах? (Ответ округлите до целых. 1 Кб = 1024 байт).

Варианты:

- a) 32 кб
- b) 64 мб
- c) 128 кб
- d) 256 кб

Ответ: _____

(время выполнения ~5 минут)

3) Рассмотрите рекурсивную функцию $F(n)$, заданную следующим образом:

$$F(n) = 1, \text{ если } n < 2;$$

$$F(n) = F(n - 1) + F(n - 2) + F(n - 2), \text{ если } n \geq 2.$$

Сколько раз будет вызвана функция $F(4)$ для вычисления результата?

Варианты:

- a) 9
- b) 15
- c) 25
- d) 5

Ответ: _____

(время выполнения ~5 минут)

4) Постройте таблицу истинности для выражения:

$$F = (A \rightarrow B) \& \neg(C \vee D)$$

(время выполнения ~7 минут)

5) Сетевой администратор анализирует трафик. Он видит пакет с IP-адресом отправителя 10.252.17.58 и адресом получателя 203.0.113.105.

Вопрос: какому классу сетей относится адрес отправителя и является ли он публичным (глобальным) или приватным (частным)?

Варианты:

- a) Класс B, публичный
- b) Класс A, приватный
- c) Класс C, публичный
- d) Класс A, публичный

Ответ: _____

(время выполнения ~3 минуты)

6) При выполнении операции "pop" в структуре данных, работающей по принципу LIFO, элемент...

Варианты:

- a) ...добавляется в начало
- b) ...возвращается из конца
- c) ...возвращается из начала
- d) ...удаляется из произвольного места

Ответ: _____

(время выполнения ~2 минуты)

2. Практико-ориентированное задание (время выполнения ~175-180 минут)

Часть А. ВЕБ-ПРОГРАММИРОВАНИЕ

7) Создайте HTML-страницу contacts.html, которая содержит:

-
- Заголовок второго уровня «Наши контакты» (выравнивание по центру).
 - Маркированный список с тремя способами связи (например, телефон, email, адрес).
 - Ссылку на главную страницу с текстом «Вернуться на главную» (адрес ссылки — index.html).
 - Используйте CSS для центрирования всего контента, задания фона цвета #e0e0e0 и изменения цвета ссылки при наведении на неё (псевдокласс :hover).

(время выполнения ~15 минут)

- 8) Добавьте на страницу contacts.html кнопку с id="fontButton" и текстом "Увеличить шрифт". При нажатии на кнопку размер шрифта всего тела страницы (тега body) должен увеличиваться на 2 пикселя от текущего значения (но не более 24px). Реализуйте это на JavaScript.

(время выполнения ~15 минут)

Часть В. АДМИНИСТРИРОВАНИЕ ОС

- 9) Напишите команду для поиска всех файлов с расширением .tmp в домашней директории пользователя (/home/) и её поддиректориях, и сохранения списка этих файлов в файл temp_files.txt.

(время выполнения ~3 минуты)

- 10) Напишите команду для завершения всех процессов с именем firefox.

(время выполнения ~2 минуты)

- 11) Напишите команду для просмотра всех запущенных служб (сервисов) в системе.

(время выполнения ~2 минуты)

- 12) Напишите скрипт для добавления правила в firewalld, которое разрешает входящие подключения по порту 8080 только для протокола tcp.

(время выполнения ~8 минут)

Часть С. ПРОГРАММИРОВАНИЕ

- 13) Напишите CLI-утилиту system_monitor.py, которая собирает и

анализирует данные о системе. Требования:

- Утилита принимает опциональный аргумент `--interval` (секунды) для периодического сбора данных (по умолчанию — однократный запуск).
- Собирает следующую информацию:
 - Загрузка CPU (в процентах).
 - Использование памяти (общая, доступная, использованная в процентах).
 - Дисковое пространство (общее, использованное, свободное для корневого раздела).
 - Сетевой трафик (отправлено/получено байт за интервал).
- Строит сводную статистику:
 - Средняя загрузка CPU за период мониторинга.
 - Максимальное использование памяти за период.
 - Топ-3 процессов по использованию CPU (имя, PID, % CPU).
 - Топ-3 процессов по использованию памяти (имя, PID, % памяти).
- Сохраняет отчёт в двух форматах: текстовый `system_report.txt` и JSON `system_report.json` в указанной папке (по умолчанию `./reports`).
- Поддерживает флаги:
 - `--duration` — общее время мониторинга в секундах (по умолчанию 60).
 - `--output-dir` — путь к папке для сохранения отчётов.
 - `--verbose` — вывод логов в консоль во время работы.
- Обработка ошибок:
 - Некорректный интервал или длительность — сообщение об ошибке.
 - Отсутствие прав доступа к данным процессов — пропуск с предупреждением.
 - Корректное завершение по `Ctrl+C`.

(время выполнения ~60-65 минут)

14) Напишите программу `library_manager.py` с классом `Library`, которая моделирует управление библиотекой и поддерживает журнал выдачи книг в файле JSON. CLI-меню позволяет добавлять книги, выдавать их читателям, принимать возвраты и отменять последнюю операцию. Требования:

- Класс `Library` с полями: `books` (словарь: `id` книги → экземпляр `Book`), `readers` (словарь: `id` читателя → экземпляр `Reader`), `transactions` (журнал операций).
- Класс `Book` с полями: `id` (строка), `title` (строка), `author` (строка), `year` (число), `is_available` (булево).

- Класс Reader с полями: id (строка), name (строка), borrowed_books (список id книг).
- Методы Library:
 - add_book(title, author, year)
 - register_reader(name)
 - borrow_book(reader_id, book_id)
 - return_book(reader_id, book_id)
- Журнал операций в файле transactions.json. Каждая запись содержит: timestamp, type (add_book, register_reader, borrow, return), details (данные операции), operation_id.
- Возможность загрузить состояние библиотеки из library_data.json при старте и сохранить при выходе.
- Команда CLI undo — отменяет последнюю операцию (например, отмена выдачи книги: книга возвращается в библиотеку, удаляется из списка читателя). При отмене — журнал дополняется записью типа undo с ссылкой на оригинальную операцию.
- Ввод должен валидировать данные (например, нельзя выдать недоступную книгу, нельзя вернуть книгу, которая не выдана), выдавать понятные ошибки.
- Тестовый набор: создание библиотеки, добавление книг и читателей, выдача книг, возврат, отмена операций.

(время выполнения ~45-50 минут)

15) Есть сетка ($N \times M$) с числами — количество монет (отрицательные числа обозначают ловушки, которые забирают монеты). Игрок стартует в левом верхнем углу (0,0) и должен дойти до правого нижнего угла ($N-1, M-1$). Можно двигаться только вниз или вправо. Задача — найти маршрут, который максимизирует количество собранных монет. Вывести суммарное количество монет, путь (координаты) и визуализацию сетки с отмеченным путём. Требования:

- Программа запрашивает размеры N (строки) и M (столбцы), где $N \geq 2$, $M \geq 2$.
- Генерирует матрицу случайных целых чисел в диапазоне $[-15, 25]$.
- Алгоритм находит максимальное количество монет от (0,0) до ($N-1, M-1$) с допустимыми ходами (вниз или вправо).
- Выводит:
 - Сгенерированную матрицу.
 - Максимальное количество монет и путь (список координат).
 - Визуализацию: матрица, где клетки пути выделены (например, символами ##).

- Алгоритм должен использовать динамическое программирование.
(время выполнения ~25-30 минут)

3. Критерии оценки

Количество часов на выполнение заданий: 3,5 ч.

| Блок | Задание | Тема | Макс. балл | Критерии оценки |
|--|---------|-------------------|------------|--|
| Теоретический блок | 1 | Системы счисления | 1 | 1 балл: Верный ответ. 0 баллов: Неверный ответ или нет ответа. |
| | 2 | Кодирование | 1 | 1 балл: Верный ответ. 0 баллов: Неверный ответ или нет ответа. |
| | 3 | Алгоритмы | 1 | 1 балл: Верный ответ. 0 баллов: Неверный ответ или нет ответа. |
| | 4 | Логика | 1 | 1 балл: Верный ответ. 0 баллов: Неверный ответ или нет ответа. |
| | 5 | Адресация в сети | 1 | 1 балл: Верный ответ. 0 баллов: Неверный ответ или нет ответа. |
| | 6 | Структуры данных | 1 | 1 балл: Верный ответ. 0 баллов: Неверный ответ или нет ответа. |
| | | ИТОГО: | 6 | |
| Практико-ориентированное задание (Часть А) | 7 | HTML/CSS | 5 | 5 баллов: Все элементы по ТЗ созданы, верстка валидна, контент центрирован. 3 балла: Элементы есть, но есть незначительные ошибки в стилях. 1 балл: Созданы только базовые элементы без стилей. 0 баллов: Задание не выполнено. |
| | 8 | JavaScript | 5 | 5 баллов: Кнопка добавлена, при нажатии фон меняется на случайный цвет. 3 балла: Кнопка есть, но цвет меняется не случайный/не меняется фон всей страницы. 1 балл: Кнопка добавлена, но функционал не работает. 0 баллов: Задание не выполнено. |

| | | | | |
|--|----|-----------------------------------|-----------|--|
| | | ИТОГО: | 10 | |
| Практико-ориентированное задание (Часть В) | 9 | Работа с файлами | 5 | 5 баллов: Команда написана верно для выбранной ОС, учитывает поддиректории. 3 балла: Команда верна, но не ищет в поддиректориях/неверный вывод. 1 балл: Предпринята попытка, но команда нерабочая. 0 баллов: Нет ответа. |
| | 10 | Управление процессами | 5 | 5 баллов: Команда написана верно и безопасно (корректное имя процесса). 3 балла: Команда верна, но есть неточности (например, не для всех процессов). 1 балл: Команда не завершает процессы корректно. 0 баллов: Нет ответа. |
| | 11 | Установленное ПО | 5 | 5 баллов: Команда выводит полный список программ/пакетов. 3 балла: Команда верна, но вывод неполный или избыточный. 1 балл: Указана неверная команда. 0 баллов: Нет ответа. |
| | 12 | Работа со скриптами, брандмауэром | 5 | 5 баллов: Скрипт/команда написаны верно для выбранной ОС, правило блокирует входящие подключения на порт 5555. 3 балла: Есть небольшие ошибки в синтаксисе или параметрах правила. 1 балл: Скрипт нерабочий или кардинально неверный. 0 баллов: Нет ответа. |
| | | ИТОГО: | 20 | |
| Практико-ориентированное задание (Часть С) | 13 | Работа с файловой системой и CLI | 20 | 20 баллов: Утилита принимает все требуемые флаги, периодически и однократно собирает все метрики, строит полную статистику (среднее CPU, макс. память, топ-процессы), сохраняет отчет в оба формата, обрабатывает ошибки |

| | | | | |
|--|----|---|----|--|
| | | | | <p>интервала/длительности и Ctrl+C, реализован verbose-режим.</p> <p>16 баллов: Все основные функции работают, но есть незначительные ошибки (например, в формате отчета, нет обработки какой-либо одной ошибки).</p> <p>12 баллов: Сбор основных метрик работает, но отсутствует построение статистики (топ процессов, средние/макс. значения) или некорректно работает один из форматов вывода.</p> <p>8 баллов: Реализован сбор только части метрик (например, нет сетевого трафика или дисков) или отсутствует поддержка ключа --interval.</p> <p>4 балла: Программа запускается и выводит данные только об одной метрике (только CPU или только память) в консоль, без сохранения в файл.</p> <p>0 баллов: Задание не выполнено или программа не запускается.</p> |
| | 14 | ООП с сериализацией и управлением состоянием через JSON | 22 | <p>22 балла: Реализованы все классы (Library, Book, Reader) с полями, все методы (add, register, borrow, return), журнал операций записывается в JSON, есть отмена последней операции (undo), реализована загрузка/сохранение состояния, ввод валидирован, обрабатываются все ошибки (недоступная книга и т.д.).</p> <p>18 баллов: Все функции работают, но есть недочеты (например, не все ошибки обрабатываются, некорректная работа undo в edge-кейсах, мелкие баги в валидации).</p> |

| | | | | |
|--|----|---|----|--|
| | | | | <p>14 баллов: Отсутствует одна из ключевых функций: либо ведение журнала операций, либо механизм отмены (undo), либо загрузка/сохранение состояния.</p> <p>10 баллов: Реализованы основные операции (добавление книг/читателей, выдача/возврат), но без журналирования, отмены операций или сохранения состояния между запусками.</p> <p>6 баллов: Реализована только часть базового функционала (например, только добавление книг в память без выдачи или без классов).</p> <p>0 баллов: Задание не выполнено или программа не запускается.</p> |
| | 15 | Алгоритмы и динамическое программирование | 22 | <p>22 балла: Программа генерирует корректную матрицу, правильно находит максимальную сумму с помощью динамического программирования, восстанавливает путь, отображает его визуально, работает с любыми размерами.</p> <p>18 баллов: Все шаги есть, но присутствуют неточности в визуализации или восстановлении пути.</p> <p>14 баллов: Программа находит максимальную сумму, но не восстанавливает путь или делает это неправильно.</p> <p>10 баллов: Реализован ввод, генерация и подсчёт суммы, но без алгоритма ДП (например, перебор или жадный алгоритм) или с критическими ошибками в реализации ДП.</p> <p>6 баллов: Есть генерация матрицы, но путь не считается.</p> |

| | | | | |
|--------------|--|---------------|------------|--|
| | | | | 0 баллов: Задание не выполнено или программа не запускается. |
| | | ИТОГО: | 64 | |
| ВСЕГО | | | 100 | |