

### **Загальні зауваження(читати уважно!).**

*Проект на Java складається з директорії, в якій містяться 3 (мінімум) підпапки:*

*папка src: файли java, де описані всі класи та їх реалізація,  
папка resources: де містяться непрограмні файли, що потрібні для виконання завдання (малюнки, файли даних і т.п.) та  
папка тестових файлів, в якому викликаються всі функції модуля та перевіряються їх робота.*

*При цьому дані повинні вводитися вручну з консолі ТА з тестового текстового файлу або бінарного файлу NZ\_test.dat(txt), в якому в зрозумілій формі містяться дані, що тестують функції. При запуску тестового файлу пропонується обрати режим тесту – введення з консолі або введення даних з тестового файлу, після цього демонструється робота програми з виводом на консоль та виводом результату у файл.*

За потребою можна підключати інші директорії в проект.

Для всіх створюваних класів створити функції введення/виведення та можливість запису в файл читання з файлу.

Оцінюється окрім коректності та відповідності коду задачі та описаним вище умовам також: акуратність коду, наявність потрібних коментарів, зрозумілість вводу/виводу, повнота тестових випадків, оптимальність алгоритмів.

Стиль коду:

<https://habr.com/ru/post/112042/>

<https://google.github.io/styleguide/javaguide.html>

**Код класів та складних методів класу повинен бути документований для javadocs!!**

Створити та прогнати файл з тестовими даними для перевірки коректності роботи створених функцій. Коментувати код згідно зразків. В коментарях файлу є прізвище розробника, дата та час виконання завдання. Додати де треба коментарі `/**коментарій*/`, та згенерувати html-файл документації.

Оцінюватися будуть наступні фактори:

1. Файли, класи та методи підготовлені згідно описаного стилю та повністю відповідають умові
  1. Коментарі та код написані згідно вказаного зразка
  2. Тести присутні для кожної функції та методу класу

3. Код працює коректно для кожного описаного тесту
4. Повнота описаних тестових випадків
5. Написаний та грамотно оформлений опис програми та спеціальних меторді, що вона використовується. Коректний пул-реквест.

## 1. Задача Гістограма

Дані для статистичної обробки поступають послідовно або пакетами. При цьому кількість даних занадто велика, щоб зерігати їх усі. Тому потрібно створити структуру, що має фіксовану довжину для зберігання великої кількості даних у вигляді гістограми частот.

Створіть структуру Гістограма (Histogram):

```
double min_hist; // Мінімальне можливе значення в структурі
double max_hist; // Максимальне можливе значення в структурі
int M; // Кількість стовбців в гістограмі (рівномірно на інтервалі [min_hist, max_hist])
int frequency[M]; // Частота - кількість елементів в кожному стовпці гістограми
```

Методи для завдання, зміни та поповнення гістограми

```
setMax(double m); // визначити максимальне можливе значення в структурі
setMin(double m); // визначити мінімальне можливе значення в структурі
setM(unsigned m); // визначити кількість стовпців на інтервалі [min_hist, max_hist]
addNumber(double x); // додати дійсне число в структуру
addbatch(double data[], int dataSize); // додати масив чисел в структуру
Передбачити можливість додання чисел з бінарного та текстових файлів (згідно вимог на початку).
```

Опишіть три варіанти додавання чисел: 1) коли числа, що не входять в інтервал [min\_hist, max\_hist] не додаються 2) коли числа, що не входять в інтервал [min\_hist, max\_hist] додаються до крайніх інтервалів 3) коли вони змінюють значення min\_hist, max\_hist

Аналіз гістограми:

```
int num(); // кількість елементів в гістограмі
int numHist(int i); // частота відповідного стовпчика
```

```
double mean(); // середнє значення гістограми
double median(); // медіана гістограми
double dev(); // середнє відхилення
```

`double variance();` // варіація гістограми

Опишіть методи для обчислення асиметрії, ексесу та ще 5 характеристик гістограми на ваш вибір.

## 2: Java Command Line (JCL)

Реалізувати консольну програму на Java, яка б представляла якусь реалізацію командного рядка, тобто могла б виконувати команди вводяться через підрядник користувачем. Під командою розуміється наступний рядок:

"Ім'я команди" "аргумент №1" "аргумент №2" ... "аргумент №N"

Код, який виконує необхідну команду користувача, повинен оформлятися як окремий Java клас. Відповідність між класом і ім'ям команди повинно здаватися в конфігураційному (XML) файлі програми. Команда під час виконання може так само взаємодіяти з користувачем використовуючи стандартний ввід / вивід.

Програма повинна підтримувати такі команди:

- `dir` - виводить список файлів в поточному каталозі
- `cd «шлях»` - перейти в директорію, шлях до якої поставлено першим аргументом
- `pwd` - вивести повний шлях до поточної директорії

1. Програма повинна підтримувати команду "!"

Ім'я\_системної\_програми аргумент№1 ... аргумент№N", яка запускає системну програму з відповідними аргументами - так само введення JCL повинен подаватися на вхід системної програми і те ж саме з її висновком

2. Програма повинна підтримувати наступний синтаксис: «команда1 аргументи && команда2 аргументи» і «команда1 аргументи || команда2 аргументи ». У першому випадку «команда2» запускається тільки якщо «команда1» виконалася успішно. У другому випадку «команда2» запускається тільки тоді, коли виконання «команда1» завершилося неуспішно

3. Програма повинна підтримувати можливість запускати команди у фоновому режимі. Для цього достатньо в кінці командного рядка ввести знак "&". Так само програма повинна підтримувати команду «jobs», яка виводить список завдань, які виконуються в тлі

4. Реалізувати можливість запускати програму в мережевому варіанті:

тобто програма відкриває порт і працювати з нею можна за допомогою звичайного telnet з віддаленій машини. Програма повинна підтримувати підключення відразу декількох користувачів, а так же команди who і write, які показують хто підключений і посилають повідомлення всім відповідно.

### **3 COLOR:**

Реалізувати кольорови системи RGBA, XYZ, HSB/HSI, CMYK. (колір задається 4-ма цілими числами в RGBA R(Red), G(Green), B(Blue), A (Opacity)). Зробіть клас Color з потрібним інтерфейсом від якого наслідники ColorRGBA, ColorHSB, ColorCMYK. Розробити схему зберігання та відображення кольору з легким переводом з одієї системи в іншу. Методи valueInt(),valueFloat()переводять колір у ціле та дійсне число.

Реалізувати введення/виведення кольору, запис масивів кольору у файл, додавання кольорів (це змішування – тобто береться середнє по кожному компоненту). Операції побітового перетину, об'єднання та XOR.

Реалізувати тип (Image)Зображення, що складається з цілих чисел m, n та двовимірної матриці m\*n з кольорів. Реалізувати тип Маска, що складається з цілих чисел m, n та двовимірної матриці m\*n з цілих чисел.

### **4. Поліном**

Створіть клас Поліном(Polynome)(2), який реалізує інтерфейс числа та ще має операції: Значення поліному, похідна, інтеграл(визначений та невизначений). На вхід конструкторів подавати масив або список чисел, або нічого (нульовий полном) або інший поліном.

Створіть класи Раціональне Ч-ло(Rational) та беззнакове ціле Unsigned та безрозмірне беззнакове BigUnsigned, що реалізують інтерфейс Числа та додати можливість створювати поліноми від цих класів. Реалізувати всі стандартні математичні операції для Rational, Unsigned та BigUnsigned. Створити функції для розв'язання лінійних та квадратних рівнянь в цих числах.

### **5. Тензор**

Створити клас-шаблон Тензор, який дозволяє вводити Тензори з консолі, та з файлів у форматах XML та JSON. Тип чисел повинен бути будь-який. Реалізувати стандартні дії з тензором та віображення й вивід тензорів на консоль та в файли.

## **6. Лабіринт**

Створіть програму, яка генерує випадковим чином пласкі та об'ємні лабіринти. Програма може відображати пласкі лабіринти та проєкції об'ємних лабіринтів. Користувач має можливість ввести координати в лабіринті та отримати найкоротший шлях виходу з даної точки.

## **7. Literature contest**

Написати сервер, який приймає від клієнтів листи, якщо лист має певну задану тему та надійшли в потрібний (заданий термін), то всі вордівські та текстові файли з нього приймають участь у літературному конкурсі.

Для цього, тексти перевіряються на належну кількість символів (більше A та менше B), та з них формують відповідні вордівські документи з однаковим (заданим) форматуванням.

Після цього клієнтам, що беруть участь в конкурсі надсилаються повідомлення у вигляді екселівських файлів з назвами розказів, де кожному розказу (які не є тими, що надіслав сам клієнт) клієнт повинен виставити оцінку від 1 до 10. Розкази можуть або відсилатись рецензентам або формується архів на сервері для скачування. При цьому кількість рецензованих кожним розказів повинна дорівнювати заданому числу N. В той же час на кожен розказ повинно бути приблизно однакова достатньо велика кількість рецензій. Після цього збираються результати та формується ітог (на зразок балів в спортивній гімнастиці). Клієнти, що не надіслали вчасно оцінок або неправильно їх виставили знімаються з конкурсу. Формується протокол з фінальними оцінками та виявленням призерів.

## **8. TextEditor: Текстовий редактор**

Створити графічний текстовий редактор, подібний до Notepad, який дозволяє створювати, редагувати, відкривати існуючий та зберігати текстові файли. Додайте туди підсвітку синтаксису та та пошук регулярних виразів у меню. Додайте також наступні функції:

- 1) Підрахувати кількість гласних та спецсимволів у тексті. Виводить звіт про це в окремий файл.
- 2) Підрахувати частотну гістограму слів та літер у тексті та вивести її у окремий файл
- 3) Зберігає відформатований текст в один з відомих відкритих форматів.

## **9. Post it Notes Program**

Створіть програму, що дозволє створювати повідомлення та працювати як будильник сигналізуючи в певний заданий час про кожне

з цих повідомлень. Зробіть локальну версію та мережеву, де повідомлення відсилається по заданому адресу через Інтернет. Зробіть її як серверне застосування, що має адмінський та клієнтський доступ. Клієнт повинен реєструватися та вказувати ці повідомлення. Адмін керує роботою клієнтів. Також програма має можливість завантажувати в певний час інформацію з мережі та відсилати її користувачу

## **10. Random Gift Suggestions**

Створити базу даних друзей(прізвища, контактні дані, сайт, мейли і т. п.). Програма слудкує за днем клендарю та визначає дати кли потрібно людину поздоровляти — день народження або загальне свято та обирає подарунок та відповідну відкритку. Програма створює письмо та відкритку з відповідним подоровленням (наприклад, дівчину з 8 березня та формує коректний персоніфікрваний текст поздоровлення та обриає доречний подарунок. При цьому шукає рекомендації згідно статі, возрасту, імені, гороскопу — пошук на відповідних сайтах). Перед відправленням поздоровлення воно виводить його у вигляді нагадування на екран та дозволяє редагувати чи вімінити його.

**11. Ciphers.** Створить клієнт-среверну програму що дозволяє обмінюватись зашифрованими повідомленнями. На початку клієнт повинен зареєструватись на сервері та обрати алгоритм шифрування (два довірних алгоритми). При цьому сервер повинен знати лише публічний ключ, а клієнт може знати і публічний і привантий. Клієнт може відправляти на сервер довільний текст, що буде клієнтом зашифрований та зберігати його там зашифрованим. Клієнт мож потім по авторизованому доступу скачувати цей текст та розшифровувати.

## **12. Web Bot**

Автоматизована програма, що через задані інтервали часу з'єднується до даного списку сайтів або знаходити ці сайти по пошуку за заданим словам, знаходить там нову інформацію по потрібним тегам та вивантажує користувачу дану інформацію. Програма здатна вивантажувати медіафайли та документ за посиланням на сайті.

Також вона повинна робити звіт про виконану роботу та знайдені сайти а також проблеми у власній роботі (цей сайт недоступний, той захищений і т.п.).

**13. Airline / Hotel Reservation System** Створіть клієнт-серверну систему, що дозволяє бронювати квитки на різні види транспорту та отелі. Врахуйте при цьому різні вартості квитків та номерів, та обмеженість місць в кожному з видів квитків та отелей. Система дозволяє працювати адміну з наповненням інформації про отелі та транспорт а для клієнтів дозволяє побирати оптимальні по вартості та/або часу маршрути в задані точки.

#### **14. Student Grade Book Application**

Створіть систему для обліку студентів, яка дозволяє зберігати та редагувати персональні дані студентів та їх успішність. Система дозволяє визначати успішність індивідуально та для кожної групи та курсу, а також по відношенню до викладача. Всі дані зберігаються в SQL сумісній БД та можуть конвертуватись пта завантажуватись в/з CSV.

#### **15. Bank Account Manager -**

Створіть клас "Account" який буду абстрактним класом для інших класів "CheckingAccount", "SavingsAccount" та "BusinessAccount". Система дозволяє керувати кредитами та поповненням разунків в стилі ATM . Система також дозволяє робити платежі та формувати випіски у документи. Зробіть цю систему клієнт-серверною. Всі дані зберігаються в SQL сумісній БД та можуть конвертуватись та завантажуватись в/з CSV.

**16. Library Catalog** Створіть клас Ресурс, що може мати наслідників типу Книга(title, page count, ISBN), Платівка, Інструмент і т. п. Створіть систему каталогів для цих ресурсів та систему для керування системою аренди для них (Бібліотека, Лізингова систем і т.ін). Система слідкує за вчасним поверненням ресурсів та начісляє штрафи у разі просрочки і т.п. Також вона може забороняти доступ до певних ресурсів і т.ін. Дані зберігаються в CSV та XML форматах.

17. **Chat** Створіть додаток для чату, який може створювати прості чати, наприклад, в Інтернет-релейному чаті (IRC) або більш прямий стиль чату, наприклад MSN. Для додаткової складності створіть власний протокол для полегшення цього чату. Створіть програму для чату, яка може створювати прості чати, наприклад, в Інтернет-релейному чаті (IRC) або більш прямий стиль чату, наприклад, MSN. Для додаткової складності створіть власний протокол, щоб полегшити цей чат. Створіть програму для чату, яка дозволяє вам безпосередньо підключитися до іншого комп'ютера за допомогою їх IP за допомогою віддалення та дозволити вашій програмі "сервер" обробляти декілька вхідних з'єднань.

18. **ImageCreator (Багатопотоковий творець мініатюр)** - обробка зображення може зайняти багато часу для деяких перетворень. Особливо, якщо зображення велике. Створіть програму для зображень, яка може приймати сотні зображень та перетворювати їх на заданий розмір у фоновому потоці, тоді як ви робите інші речі. Для додаткової складності потрібно переробити розмір однієї нитки, мати ще одну для перейменування ескізів тощо.

19. **Online White Board** - Створіть додаток, який дозволяє вам та друзям співпрацювати на білій дошці в Інтернеті. Малюйте малюнки, пишіть замітки та використовуйте різні кольори, щоб обробляти ідеї для проектів. Спробуйте завантажувати картинки. Спробуйте зберегти файли в один з відомих форматів.

20. **Page Scraper** - створення програми, яка підключається до сайту та витягує всі посилання чи зображення та зберігає їх у списку. Для додаткової складності впорядковуйте індексований вміст і не дозволяйте дублікатів. Нехай він помістить результати в індекс, що легко шукається. File Fileer - програма, яка може завантажувати різні об'єкти на сторінку, включаючи відеопотоки або всі файли на сторінці. Чудово підходить для сторінок з великою кількістю посилань для завантаження. Monitor Bandwidth Monitor - невелика утилітна програма, яка відстежує кількість даних, які ви завантажили та завантажили з мережі в ході вашого поточного онлайн-сеансу. Подивіться, чи



зможете ви дізнатися, у які періоди дня ви все більше і менше використовуєте, і створити звіт або графік, який це показує.

**21. Quiz Maker** - Складіть програму, яка містить різноманітні запитання, утворюючи файл, вибраний випадковим чином, і збирає вікторину для студентів. Запитання повинні бути завантажені в CSV та JSON форматах. Та ці бази запитань можуть підлягати змінам в режимі адміністратора. Кожна вікторина може бути різною, після чого читається ключ, щоб оцінити вікторини та відправляється по пошті повідомлення про результати.

**22. Провідник файлів** - Створіть власну програму провідника Windows, але з додатковими функціями, кращим пошуком, новими піктограмами та іншими видами перегляду. Наприклад, якщо користувач вводить `myimage ###.Jpg`, він буде перейменовувати всі файли з "мінімальним" з трьох чисел, наприклад `"myimage001.jpg"`, `"myimage145.jpg"` або навіть `"myimage1987.jpg"` з 1987 року, принаймні три numbers. **File Copy Utility** - Створіть утиліту, яка може робити масове копіювання файлів та резервне копіювання інших файлів. Ця програма використовується для перегляду різних файлів зображень на вашому комп'ютері від PNG, GIF, JPG до BMP, TIFF тощо.

**23. Віддалений інструмент SQL** - утиліта, яка може виконувати запити на віддалених серверах з вашого локального комп'ютера через Інтернет. Він повинен зайняти віддалений хост, ім'я користувача та пароль, запустити запит та повернути результати. Забезпечте безпечний адмінський та користувацький доступ. Створіть утиліту (Генератор звітів), яка генерує звіт на основі отриманих з інтернету таблиць. Таблиці шукаються на деякому сайті. Програма створює звіти про продажі на основі таблиць із деталями замовлення / замовлення або підсумовує дні в поточному сайті.

**24. Система планування подорожей** - Створіть систему, яка дозволяє користувачам складати власний невеликий маршрут подорожей та слідкувати за умовами авіакомпанії / готелів, визначними місцями,

бюджетом та розкладом. Всі дані зберігаються в базах даних SQL. Виведення — графічно на сайті.

**25. Screen Capture Program** - Зробіть утиліту, яка просто захопить кадр із веб-камери. Для додаткової складності можна також вбудувати функціональність електронної пошти. Ця програма використовується також дозволяє програвати відеозображення.

**26. Додаток для спільного використання файлів P2P** - Створіть таку програму, як LimeWire, FrostWire, Bearshare або додаток у стилі торрент. При цьому створюється Packet Sniffer - корисна програма, яка буде читати пакети, що надходять і виходять з машини, а також супутню інформацію, наприклад призначення та розмір корисного навантаження.

**27. Перевірка пошти (POP3 / IMAP)** - Користувач вводить різні дані облікового запису, включаючи веб-сервер та IP, тип протоколу (POP3 або IMAP), і програма перевірятиме наявність електронної пошти на декількох акаунтах у визначений інтервал. На деякі з емейлів встановлюється спам-фільтр, а на деякі автовідповідач. Опцій автовідповідача прописуються в JSON файлах.

**28. Gameserver** Створіть ігровий сервер з візуальною частинкою, що дозволяє грати у Hangman та Кросворд. Hangman: Випадково виберіть слово з файлу, щоб користувач вгадав символи слова. Для кожного персонажа вони здогадуються, що це не в слові, нехай він намалює іншу частину людини, що висить у петлі. Якщо картина завершена до того, як вони відгадають усіх персонажів, вони впрограють.

Кросворд - Створіть кросворд, який поєднує слова разом на загальних літерах. Надайте список підказок для кожного слова та дозвольте користувачеві вводити слова, доки не буде заповнено всю кросворд.

**29. StyleMaker** - Ви бачили коли-небудь повідомлення на веб-дошці, де хтось створив згенерований малюнок? Подивіться, чи можете ви створити програму, яка дозволяє користувачеві задавати фон, текст, кольори та вирівнювання, щоб зробити власні підписи або

користувацькі смуги. Додайте власний логотип чи текст злегка на задньому плані, щоб ніхто не міг просто вкрати вашу графіку з вашого сайту. Складіть програму, яка додасть цей водяний знак до малюнка.

Візуальні застосування.

Створити на базі сокетів клієнт /серверний візуальний додаток:

30. **Чат 1.** Клієнт посилає через сервер повідомлення іншому клієнту та клієнт посилає через сервер повідомлення іншому клієнту, вибранному зі списку. Клієнт посилає через сервер повідомлення, яке отримують всі клієнти. Список клієнтів зберігається на сервері у файлі.

31. **Сонети.** Клієнт при зверненні до сервера отримує випадково обраний сонет Шекспіра з файлу (текстовий та XML). Сервер розсилає повідомлення обраним зі списку клієнтам. Перелік зберігається в файлі JSON. Сервер розсилає повідомлення в певний час певним клієнтам. Сервер розсилає повідомлення тільки тим клієнтам, які в даний момент знаходяться в on-line.

32. **Чат 2.** Сервер розсилає всім клієнтам інформацію про клієнтів які йшли в чат і залишили його. Також можна переслати повідомлення кожному з клієнтів. Крім того, є база зображень яка відображається клієнтам. Клієнт може обрати зображення зі списку і пересилає його іншому клієнту через сервер.

33. **Gamesrever-2** Ігри по мережі на безрозмірному (великому) полі:

а) Хрестики-нулики 2. Для перемоги необхідно вибудувати п'ять в один ряд або діагональ.

б) Ігра- аналог "Го". Два гравці ходять по черзі — змінюючи колір вузла клітинки При цьому якщо один з гравців може оточити (кожний вузол приєднується до одного з сусудів-вузлів, що є його коольору) вузли ішого гравця — то він їх закрашує влсним коліром(бере в полон). Перемагає той хо через заданий час взяв в полон більше чужих вузлів.

34. **Проксі** Написати програму, що сканує мережу в зазначеному діапазоні IP адресів на наявність активних комп'ютерів. Програма повинна приймати повідомлення від будь-якого клієнта, працюючого на протоколі TCP, і відсилати їх на відповідний сервер. При передачі змінювати повідомлення.