OCC 12

UNIVERSITI MALAYA
UNIVERSITI MALAYA

PEPERIKSAAN IJAZAH SARJANA MUDA SAINS KOMPUTER
EXAMINATION FOR THE DEGREE OF BACHELOR OF COMPUTER SCIENCE

SESI AKADEMIK 2024/2025 : SEMESTER 1
ACADEMIC SESSION 2024/2025 : SEMESTER 1

WIX1002 : Asas-Asas Pengaturcaraan
Fundamentals of Programming

Februari 2025
February 2025

Masa: 3 jam 30 minit
Time: 3 hours 30 minutes

ARAHAN KEPADA CALON:
INSTRUCTIONS TO CANDIDATES:

Calon dikehendaki menjawab **SEMUA** soalan.
*Candidates should answer **ALL** questions.*

(Kertas soalan ini mengandungi 5 soalan dalam 16 halaman yang dicetak)
(This question paper consists of 5 questions on 16 printed pages)

1.

a) Tulis satu program untuk menukar sistem nombor perpuluhan kepada perduaan menggunakan operator modulus dan bahagi. Pelajar tidak dibenarkan menggunakan API sedia ada seperti kelas Math, Collection dan sebagainya.

Write a program to convert decimal to binary number system using modulus and division operators. Students are not allowed to use predefined APIs such as Math, *or Collection classes and such.*

Contoh Output:
Sample Output:

```
Enter the Decimal Number: 10
Binary Number : 1010
```

(5 markah/*marks*)

b) Tulis satu program Java menggunakan penyataan *switch-case* untuk menilai gred pelajar berdasarkan skor angka. Program tersebut hendaklah:

i. Meminta pelajar untuk memasukkan 5 markah kursus dengan nombor perpuluhan antara 0 dan 100.

ii. Kira purata bagi markah 5 kursus.

iii. Gunakan pernyataan *switch-case* untuk menentukan gred purata yang boleh menjadi nombor perpuluhan; sekiranya purata ialah perpuluhan, kekalkan ia seperti sedia ada. Jangan bundarkan purata. Nyatakan gred mengikut skala berikut:

A: 90–100
B: 80–89
C: 70–79
D: 60–69
F: 0–59

Write a Java program using a switch-case statement to evaluate a student's grade based on a numeric score. The program should:

i. Prompt the student to enter 5 courses score with a decimal number between 0 and 100.

ii. Calculate the average for the 5 courses score.

iii. *Use a switch-case statement to determine the average grade which can be a decimal number; in case the average is decimal keep it as it is. Do not round the average. Specify the grade according to the following scale:*

A: 90–100 B: 80–89 C: 70–79 D: 60–69 F: 0–59

Contoh Output:
Sample Output:

```
Enter the scores for 5 courses (each between 0 and 100):
Course 1 score: 980
Invalid score. Please enter a score between 0 and 100.
Course 1 score: 98.9
Course 2 score: 89.9
Course 3 score: 162
Invalid score. Please enter a score between 0 and 100.
Course 3 score: 79.8
Course 4 score: 86.7
Course 5 score: 86.7
Average score: 88.4
Grade: B
```

(5 markah/*marks*)

2.

a) Tulis satu kaedah Java bernama **findLargestEven** yang menjana satu tatasusunan 10 integer rawak antara 1 dan 100 menggunakan kelas Random. Kaedah ini kemudian perlu mencari dan memulangkan nombor genap terbesar dalam array tersebut. Jika tiada nombor genap ditemui, kaedah ini perlu memulangkan -1. Laksanakan penyelesaian ini menggunakan array, kaedah, dan kawalan aliran.

Nota: Pelajar tidak dibenarkan menggunakan API sedia ada seperti Arrays.sort(), Streams, atau kelas Collection untuk mengendalikan atau menyusun tatasusunan.

*Write a Java method named **findLargestEven** that generates an array of 10 random integers between 1 and 100 using the Random class. The method should then find and return the largest even number in the array. If no even numbers are found, the method should return -1. Implement the solution using arrays, methods, and flow control.*

***Note:** Students are not allowed to use predefined APIs such as* Arrays.sort*(), Streams, or Collection classes for sorting or handling the array.*

Contoh Output:
Sample Output:

```
The generated array is: [21, 34, 67, 82, 14, 71, 53, 45, 90, 29]
The largest even number is: 90
```

(5 markah/*marks*)

b) Tulis jawapan kepada dua soalan berikut berkenaan dengan kod sumber yang diberikan.

i. Terangkan pelaksanaan program yang diberikan di bawah.
ii. Nyatakan output program di bawah apabila ia dijalankan, dan terangkan **mengapa?**

Write the answers for the following two questions with respect to the given source code.

i. Explain the execution of the given program below.
*ii. Specify the output of the following program when it runs, and explain **why**?*

```
public class FlowControlTest {
    public static void main(String[] args) {
        int x = 15;
        int y = 5;
        int z = 10;

        if (x > y) {
            if (y > z) {
                System.out.println("Condition 1");
            } else {
                if (x < z) {
                    System.out.println("Condition 2");
                } else {
                    System.out.println("Condition 3");
                }
            }
        } else {
            if (z > x) {
                if (z > y) {
                    System.out.println("Condition 4");
                } else {
                    System.out.println("Condition 5");
                }
            } else {
                System.out.println("Condition 6");
            }
        }
    }
}
```

(5 markah/*marks*)

3 Anda dikehendaki mereka bentuk sistem pengurusan perpustakaan. Sistem ini perlu mengendalikan pelbagai jenis bahan perpustakaan dan memastikan proses peminjaman berjalan lancar dan bebas daripada kesilapan.

You are required to design a library management system. The system needs to handle different types of library materials and ensure that the borrowing process is smooth and error-free.

a) Takrifkan kelas **LibraryMaterial** untuk mewakili bahan perpustakaan. Kelas ini sepatutnya mempunyai yang berikut:

i. Tajuk bahan.

ii. Pengarang bahan.

iii. Adakah bahan itu tersedia untuk dipinjam.

iv. Tarikh akhir untuk bahan tersebut dipulangkan jika dipinjam. Bahan perpustakaan boleh dipinjam selama maksimum 14 hari.

v. Pembina untuk mengasal tajuk, pengarang, ketersediaan bahan dan tarikh pemulangan bahan.

vi. Kaedah **borrow()** untuk dilaksanakan apabila bahan dipinjam daripada perpustakaan.

*Define a class **LibraryMaterial** to represent a library item. This class should have the following:*

i. The title of the item.

ii. The author of the item.

iii. Is the item available to be borrowed.

iv. The due date for the item to be returned if it is borrowed. Library materials can be borrowed for a maximum of 14 days.

v. A constructor to initialize the item's title, author, availability, and return date.

*vi. A **borrow()** method to be executed when the item is borrowed from the library.*

(3 markah/*marks*)

b) Kembangkan kelas **LibraryMaterial** untuk menyokong pelbagai jenis bahan.

i. Subkelas **DVD** yang mewarisi daripada **LibraryMaterial**. Subkelas ini harus mempunyai atribut *duration* untuk menerangkan tempoh masa larian item. Dalam subkelas ini, gantikan kaedah **borrow()** untuk mengehadkan tempoh peminjaman, sebagai contoh, "DVD hanya boleh dipinjam selama maksimum lima hari".

ii. Subkelas ***Majalah*** yang mewarisi daripada **LibraryMaterial**. Dalam subkelas ini, ganti kaedah **borrow()** untuk memaparkan mesej "Majalah tidak boleh dipinjam" dan menghalang item majalah daripada dipinjam.

*Extend the **LibraryMaterial** class to support different types of materials.*

*i. **DVD** subclass that inherits from **LibraryMaterial**. This subclass should have an attribute **duration** to describe the runtime length of the item. In this subclass, override the **borrow()** method to limit borrowing duration, for example, "DVDs can only be borrowed for a maximum of five days".*

*ii. **Magazine** subclass that inherits from **LibraryMaterial**. In this subclass, override the **borrow()** method to display the message "Magazines cannot be borrowed" and prevents the magazine item from being borrowed.*

(4 markah/*marks*)

c) Tambah pengendalian pengecualian pada kaedah **borrow()**, sebagai contoh, dengan menambahkan ***AlreadyBorrowedException***, yang sepatutnya melontar mesej jika pengguna cuba meminjam bahan yang telah dipinjam.

Tunjukkan perkara berikut dalam kaedah **main()**:

i. Mencipta contoh **LibraryMaterial**, **DVD** dan ***Majalah***.

- ***LibraryMaterial*** - *Tajuk : "The Great Gatsby"; Pengarang bahan : "F. Scott Fitzgerald"*
- ***DVD*** - *Tajuk : "Inception", Pengarang bahan : "Christopher Nolan", Durasi: "148"*
- ***Majalah*** – *Tajuk: "National Geographic", Pengarang bahan: "Various authors"*

ii. Panggil **borrow()** untuk setiap contoh.

iii. Tunjukkan contoh **LibraryMaterial** yang berjaya dipinjam dengan mesej "Bahan *tajuk* oleh *pengarang* berjaya dipinjam dan harus dipulangkan selewat-lewatnya pada *tarikhakhir*".

iv. Tunjukkan contoh **LibraryMaterial** tidak berjaya dipinjam dengan mesej "Bahan *tajuk* oleh *pengarang* pada masa ini tidak tersedia dan sepatutnya tersedia selepas *tarikhakhir*".

v. Tunjukkan contoh **DVD** yang berjaya dipinjam dengan tarikh pemulangan yang betul.

vi. Tunjukkan contoh **Majalah** tidak berjaya dipinjam dengan mesej yang betul.

*Add exception handling to the **borrow()** method, for example, by adding an **AlreadyBorrowedException**, which should throw a message if a user tries to borrow a material that is already being borrowed.*

*Demonstrate the following in the **main()** method:*

*i. Create instances of **LibraryMaterial**, **DVD**, and **Magazine** with the following information :*

- ***LibraryMaterial** - Title : "The Great Gatsby"; Author : "F. Scott Fitzgerald"*
- ***DVD** - Title : "Inception", Author : "Christopher Nolan", Duration:"148"*
- ***Magazine** - Title: "National Geographic", Author: "Various authors"*

*ii. Call **borrow()** for each instance.*

*iii. Show an instance of **LibraryMaterial** being successfully borrowed with the message "The item *title* by *author* is successfully borrowed and should be returned by *duedate*".*

*iv. Show an instance of **LibraryMaterial** being unsuccessfully borrowed with the message "The item *title* by *author* is currently unavailable and should be available after *duedate*".*

*v. Show an instance of **DVD** being successfully borrowed with the correct return date.*

vi. *Show an instance of **Magazine** being unsuccessfully borrowed with the correct message.*

(3 markah/*marks*)

Contoh Output:
Sample Output:

```
Material borrowed successfully.
Due date for return: [Today's Date + 14days]
The item "The Great Gatsby" by F. Scott Fitzgerald is successfully borrowed and
should be returned by [Today's Date + 14days].

The item "The Great Gatsby" by F. Scott Fitzgerald is currently unavailable and
should be available after [Today's Date + 14days].

DVD borrowed successfully.
Due date for return: [Today's Date + 5days]
The DVD "Inception" by Christopher Nolan is successfully borrowed and should be
returned by [Today's Date + 5days].

Magazines cannot be borrowed.
```

4. Tulis satu program yang membaca kandungan fail teks yang dibekalkan (**"Q4_input.txt"**), melakukan beberapa operasi pemprosesan teks dan menyimpan output ke fail. Anda dikehendaki menulis:

*Write a program that reads the content of the provided text file (**"Q4_input.txt"**), performs several text processing operations, and saves the output to a file. You are required to write:*

a) Kelas *TextManipulator*

i. Satu pembina hujah yang menerima satu tatasusunan String.

ii. Kaedah pembersihan teks: Alih keluar simbol yang tidak diingini.

iii. Kaedah terbalik teks: Terbalikkan teks untuk membentuk perkataan yang betul.

iv. Kaedah penyusunan semula teks: Susun semula kedudukan teks ke urutan yang betul.

v. Kaedah paparan: Mencetak kandungan tatasusunan.

vi. Kaedah *get*: Kembalikan tatasusunan.

A TextManipulator class

i. *Single-argument constructor that accepts a String array.*

ii. *Text cleaning method: Remove unwanted symbols.*

iii. *Text reverse method: Reverse the texts to form the correct words.*

iv. *Text rearrange method: Rearrange the position of the texts to the correct sequence.*

v. *Display method: Prints the contents of the array.*

vi. *Get method: Return the array.*

(6 markah/*marks*)

b) Program ujian yang:

i. Membaca dan memaparkan kandungan fail input.

ii. Membina satu objek kelas StringManipulator.

iii. Lakukan semua operasi prapemprosesan yang diperlukan.

iv. Simpan hasil akhir ke fail output yang dipanggil "**output.txt**".

v. Sertakan pengendalian ralat apabila diperlukan menggunakan pernyataan *try-catch*.

vi. Tidak dibenarkan menggunakan ArrayList.

A test program that:

i. *Reads and display the contents of the input file.*

ii. *Creates an object of class StringManipulator.*

iii. *Performs all the required preprocessing operations.*

iv. *Saves the final outcome to an output file called "**output.txt**".*

v. *Includes error handling whenever required using try-catch statement.*

vi. *The use of ArrayList is prohibited.*

(4 markah/*marks*)

Fail input yang diberikan ("Q4_input.txt"):

Provided Input file ("Q4_input.txt"):

```
"GN#IM~~MA#RGO@RP"
"@F~O"
"S@LAT##NE~M#ADN@UF"
```

Contoh output:
Sample output:

```
Original text:
"GN#IM~~MA#RGO@RP"
"@F~O"
"S@LAT##NE~M#ADN@UF"

Cleaned text:
GNIMMARGORP
FO
SLATNEMADNUF

Reversed text:
PROGRAMMING
OF
FUNDAMENTALS

Arranged text:
FUNDAMENTALS
OF
PROGRAMMING
```

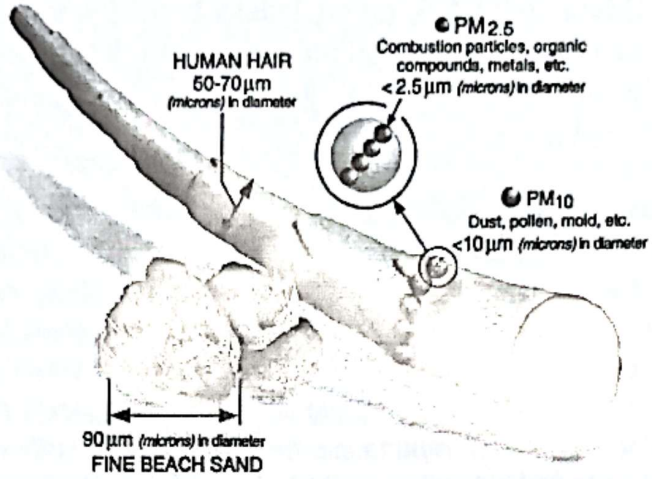
Contoh fail keluaran ("output.txt"):
Sample output file ("output.txt"):

```
FUNDAMENTALS
OF
PROGRAMMING
```

5. Bahan zarah, juga dipanggil pencemaran zarah, merujuk kepada campuran zarah pepejal dan titisan cecair yang terdapat di udara. Walaupun sesetengah zarah, seperti habuk, kotoran, jelaga, atau asap, cukup besar untuk dilihat dengan mata kasar, yang lain adalah sangat kecil sehingga ia hanya boleh dikesan dengan mikroskop elektron. **$PM_{2.5}$**, atau zarah berdiameter 2.5 mikrometer atau lebih kecil, ialah sejenis pencemar udara yang terkenal dengan risiko kesihatannya yang ketara. Saiznya yang kecil membolehkannya menembusi jauh ke dalam paru-paru dan juga memasuki aliran darah, menimbulkan kebimbangan kesihatan yang serius. $PM_{2.5}$ terdiri daripada zarah ultra halus ini dan selalunya termasuk bahan kompleks seperti bahan kimia organik, logam dan habuk.

*Particulate matter, also called particle pollution, refers to a mixture of solid particles and liquid droplets found in the air. While some particles, like dust, dirt, soot, or smoke, are large enough to be seen with the naked eye, others are so tiny that they can only be detected with an electron microscope. **$PM_{2.5}$**, or particulate matter with a diameter of 2.5 micrometers or smaller, is a type of air pollutant known for its significant health risks. Its tiny size allows it to penetrate deep into the lungs and even enter the bloodstream, posing serious health*

concerns. $PM_{2.5}$ consists of these ultra-fine particles and often includes complex substances such as organic chemicals, metals, and dust



Rajah 1: Perbandingan saiz untuk zarah PM (sumber: Agensi Perlindungan Alam Sekitar Amerika Syarikat - https://www.epa.gov)

Figure 1: Size comparisons for PM particles (source: United States Environmental Protection Agency - https://www.epa.gov)

Pendedahan kepada $PM_{2.5}$ dikaitkan dengan pelbagai isu kesihatan, terutamanya yang menjejaskan sistem pernafasan dan kardiovaskular. Ia boleh memburukkan asma, mencetuskan serangan jantung, dan menyumbang kepada penyakit paru-paru kronik dan penurunan fungsi paru-paru. Pendedahan berpanjangan kepada tahap tinggi $PM_{2.5}$ juga telah dikaitkan dengan peningkatan risiko keadaan kronik dan kematian pramatang. Oleh itu, pemantauan dan pengurusan tahap $PM_{2.5}$ adalah penting untuk menjaga kesihatan awam, kerana mengurangkan pendedahan boleh meningkatkan kesihatan keseluruhan dan jangka hayat dengan ketara, terutamanya di kawasan bandar berpenduduk padat dengan trafik tinggi dan pencemaran industri.

Exposure to $PM_{2.5}$ is associated with numerous health issues, particularly affecting the respiratory and cardiovascular systems. It can worsen asthma, trigger heart attacks, and contribute to chronic lung disease and decreased lung function. Prolonged exposure to high levels of $PM_{2.5}$ has also been linked to an increased risk of chronic conditions and premature death. Therefore, monitoring and managing $PM_{2.5}$ levels is essential for safeguarding public health, as reducing exposure can significantly enhance overall health and life expectancy, especially in densely populated urban areas with high traffic and industrial pollution.

AirNow (https://www.airnow.gov) ialah perkongsian Agensi Perlindungan Alam Sekitar A.S., Pentadbiran Lautan dan Atmosfera Kebangsaan (NOAA), Perkhidmatan Taman Negara, NASA, Pusat Kawalan Penyakit, puak, negeri dan agensi kualiti udara tempatan. Tapak web dan aplikasi AirNow menyerlahkan kualiti udara di kawasan setempat tertentu terlebih dahulu, sambil tetap

memberikan maklumat kualiti udara pada pandangan negeri, negara dan dunia. Peta interaktif AirNow malah membenarkan pengguna mengezum keluar untuk mendapatkan gambaran besar atau menggerudi ke bawah untuk melihat data bagi pemantauan kualiti udara tunggal. AirNow melaporkan kualiti udara menggunakan **Indeks Kualiti Udara (AQI)** A.S. rasmi, indeks berkod warna yang direka untuk berkomunikasi sama ada kualiti udara sihat atau tidak sihat. Apabila anda mengetahui AQI di kawasan anda, anda boleh mengambil langkah untuk melindungi kesihatan anda.

AirNow *(https://www.airnow.gov) is a partnership of the U.S. Environmental Protection Agency, National Oceanic and Atmospheric Administration (NOAA), National Park Service, NASA, Centers for Disease Control, and tribal, state, and local air quality agencies. AirNow's website and app highlights air quality in a given local area first, while still providing air quality information at state, national, and world views. AirNow's interactive map even lets users zoom out to get the big picture or drill down to see data for a single air quality monitoring. AirNow reports air quality using the official U.S.* ***Air Quality Index (AQI)****, a color-coded index designed to communicate whether air quality is healthy or unhealthy. When you know the AQI in your area, you can take steps to protect your health.*

Agensi di seluruh negara menghantar data pemantauan mereka kepada AirNow untuk paparan. Jabatan Negara menyediakan data daripada **Kedutaan dan Konsulat A.S.** untuk memaklumkan kakitangan dan rakyat di luar negara. Rajah 2 di bawah menunjukkan sampel ukuran yang dikumpul setiap jam di Kedutaan A.S. di Malaysia untuk tahun 2024.

Agencies all over the country send their monitoring data to AirNow for display. The Department of State provides data from ***U.S. Embassies and Consulates*** *to inform personnel and citizens overseas. Figure 2 below is showing a sample of the measurement collected hourly at the U.S. Embassy in Malaysia for the year 2024.*

KualaLumpur_PM2.5_2024_YTD

Site	Parameter	Date (LT)	Year	Month	Day	Hour	NowCast Conc.	AQI	AQI Category	Raw Conc.	Conc. Unit	Duration	QC Name
Kuala Lumpur	PM2.5 - Principal	2024-01-01 01:00 AM	2024	1	1	1	32.3	94	Moderate	50.0	UG/M3	1 Hr	Valid
Kuala Lumpur	PM2.5 - Principal	2024-01-01 02:00 AM	2024	1	1	2	29.5	89	Moderate	27.0	UG/M3	1 Hr	Valid
Kuala Lumpur	PM2.5 - Principal	2024-01-01 03:00 AM	2024	1	1	3	26.7	84	Moderate	24.0	UG/M3	1 Hr	Valid
Kuala Lumpur	PM2.5 - Principal	2024-01-01 04:00 AM	2024	1	1	4	24.3	79	Moderate	22.0	UG/M3	1 Hr	Valid
Kuala Lumpur	PM2.5 - Principal	2024-01-01 05:00 AM	2024	1	1	5	18.6	69	Moderate	13.0	UG/M3	1 Hr	Valid
Kuala Lumpur	PM2.5 - Principal	2024-01-01 06:00 AM	2024	1	1	6	15.3	63	Moderate	12.0	UG/M3	1 Hr	Valid
Kuala Lumpur	PM2.5 - Principal	2024-01-01 07:00 AM	2024	1	1	7	14.6	61	Moderate	14.0	UG/M3	1 Hr	Valid
Kuala Lumpur	PM2.5 - Principal	2024-01-01 08:00 AM	2024	1	1	8	12.8	58	Moderate	11.0	UG/M3	1 Hr	Valid
Kuala Lumpur	PM2.5 - Principal	2024-01-01 09:00 AM	2024	1	1	9	14.9	62	Moderate	17.0	UG/M3	1 Hr	Valid
Kuala Lumpur	PM2.5 - Principal	2024-01-01 10:00 AM	2024	1	1	10	15.9	64	Moderate	17.0	UG/M3	1 Hr	Valid
Kuala Lumpur	PM2.5 - Principal	2024-01-01 11:00 AM	2024	1	1	11	12.9	58	Moderate	10.0	UG/M3	1 Hr	Valid
Kuala Lumpur	PM2.5 - Principal	2024-01-01 12:00 PM	2024	1	1	12	12.4	57	Moderate	12.0	UG/M3	1 Hr	Valid
Kuala Lumpur	PM2.5 - Principal	2024-01-01 01:00 PM	2024	1	1	13	11.7	56	Moderate	11.0	UG/M3	1 Hr	Valid
Kuala Lumpur	PM2.5 - Principal	2024-01-01 02:00 PM	2024	1	1	14	10.3	53	Moderate	9.0	UG/M3	1 Hr	Valid
Kuala Lumpur	PM2.5 - Principal	2024-01-01 03:00 PM	2024	1	1	15	8.1	45	Good	6.0	UG/M3	1 Hr	Valid
Kuala Lumpur	PM2.5 - Principal	2024-01-01 04:00 PM	2024	1	1	16	8.5	47	Good	9.0	UG/M3	1 Hr	Valid
Kuala Lumpur	PM2.5 - Principal	2024-01-01 05:00 PM	2024	1	1	17	8.5	47	Good	-999.0	UG/M3	1 Hr	Missing

Rajah 2: Contoh fail data untuk PM2.5 (sumber: AirNow - https://www.airnow.gov/international/us-embassies-and-consulates)

Figure 2: Sample data file for $PM_{2.5}$ (source: AirNow - https://www.airnow.gov/international/us-embassies-and-consulates)

Setiap fail data mempunyai 14 lajur dan beberapa ribu rekod. Setiap rekod menunjukkan tahap kepekatan zarah (dilabel sebagai "`Raw Conc.`") dalam µg/m^3. "`NowCast Conc`" ialah purata 12 jam sebelumnya atau tahap kepekatan, dan ia digunakan untuk mengira nilai "`AQI`" menggunakan formula yang ditakrifkan oleh **Agensi Perlindungan Alam Sekitar (EPA) A.S.** "`AQI Category`" ditentukan berdasarkan Asas AQI yang ditunjukkan dalam Jadual 1. Setiap rekod akan melalui proses kawalan kualiti untuk menentukan kesahihan pengukuran, seperti yang ditunjukkan dalam lajur "`QC Name`".

*Each of the data file has 14 columns and few thousands of records. Each record shows the particles concentration level (labelled as "`Raw Conc.`") in µg/m^3. The "`NowCast Conc.`" is an average of the previous 12 hours or the concentration level, and it's used to calculate the "`AQI`" value using the formula defined by **U.S. Environmental Protection Agency (EPA)**. The "`AQI Category`" is determined based on the AQI Basis shown in Table 1. Each of the record will go through a quality control process to determine the validity of the measurement, as shown in column "`QC Name`".*

Jadual 1: Asas AQI untuk Pencemaran Ozon dan Zarah (sumber: AirNow - https://www.airnow.gov/aqi/aqi-basics/)

Table 1: AQI Basics for Ozone and Particle Pollution (source: AirNow - https://www.airnow.gov/aqi/aqi-basics/)

Daily AQI Color	Levels of Concern	Values of Index	Description of Air Quality
Green	Good	0 to 50	Air quality is satisfactory, and air pollution poses little or no risk.
Yellow	Moderate	51 to 100	Air quality is acceptable. However, there may be a risk for some people, particularly those who are unusually sensitive to air pollution.
Orange	Unhealthy for Sensitive Groups	101 to 150	Members of sensitive groups may experience health effects. The general public is less likely to be affected.
Red	Unhealthy	151 to 200	Some members of the general public may experience health effects; members of sensitive groups may experience more serious health effects.
Purple	Very Unhealthy	201 to 300	Health alert: The risk of health effects is increased for everyone.
Maroon	Hazardous	301 and higher	Health warning of emergency conditions: everyone is more likely to be affected.

Daily AQI Color	Levels of Concern	Values of Index	Description of Air Quality
Maroon	Hazardous	301 and higher	Health warning of emergency conditions: everyone is more likely to be affected.

Tugas anda ialah menulis program analisis ringkas untuk saintis data di Pusat Perubatan UM untuk diguna pakai dalam program pengurusan pesakit asma mereka. Anda harus menentukan kelas `Mon_Site` untuk mewakili setiap tapak yang dipantau AirNow, dengan atribut berikut:

- `siteName` (String): Nama tapak yang dipantau.
- `year` (String): Tahun data telah direkodkan.
- `recordCount` (int): Jumlah bilangan rekod yang telah dikumpul.
- `records` (String[][]): Rekod sebenar telah dikumpul.

Your task is to write a simple analysis program for the data scientist in UM Medical Centre to be adopted in their asthma patient management program. You should define a `Mon_Site` class to represent each of the monitored sites of the AirNow, with the following attributes:

- *`siteName` (String): The name of the monitored site.*
- *`year` (String): The year of the data been recorded.*
- *`recordCount` (int): Total number of the records been collected.*
- *`records` (String[][]): The actual records been collected.*

Program anda harus melaksanakan tiga fungsi:

Your program should perform three functions:

a) Muatkan semua fail CSV dari direktori tapak.

Petunjuk: Kita tidak tahu berapa banyak fail wujud dalam direktori itu. Gunakan kaedah dalam objek Fail Java untuk mendapatkan kesemuanya. Kita hanya berminat pada lajur `date`, `NowCast Conc.`, `AQI`, `AQI Category` dan `QC Name`. Oleh itu, anda tidak perlu memuatkan kesemua 14 atribut untuk setiap rekod.

Load all CSV files from the site directory.

Hint: We do not know how many files exist in that directory. Use the method in Java File object to retrieve all of them. We only interested in the `date`, `NowCast Conc.`, `AQI`, `AQI Category` and `QC Name` columns. Thus, you do not need to load all of the 14 attributes for each of the records.

b) Cetak statistik *QC Name*, seperti yang ditunjukkan dalam output sampel.

Petunjuk: terdapat empat kemungkinan nilai *QC Name*, iaitu `Valid`, `Invalid`, `Missing` and `Suspect`. Data hendaklah diisih mengikut nama tapak, mengikut susunan abjad.

Print the QC Name statistic, as shown in the sample output.

Hint: there are four possible QC Name values, i.e. `Valid`, `Invalid`, `Missing` and `Suspect`. The data should be sorted by the name of the site, in alphabetical order.

c) Hasilkan laporan untuk setiap tapak yang dipantau. Jangan sertakan rekod tersebut dengan nilai `QC Name`, `Invalid`, `Missing` dan `Suspect`, atau nilai `AQI` "`N/A`". Simpan laporan ke dalam fail teks masing-masing dengan nama tapak dan tahun telah direkodkan, dengan sambungan "*`.site`*", seperti yang ditunjukkan dalam contoh output.

Petunjuk: Rajah 3 menunjukkan dua rekod yang perlu dikeluarkan daripada statistik, iaitu rekod kedua (nilai QC Name *`Missing`*) dan rekod ketiga (nilai AQI ialah "*`N/A`*").

Generate a report for each of the monitored sites. Do not include those records with `QC Name` values `Invalid`, `Missing` and `Suspect`, or `AQI` value "`N/A`". Save the report into a respective text file with the name of the site and the year been recorded, with "`.site`" extension, as shown in the sample output.

Hint: Figure 3 shows two records that need to be removed from the statistic, i.e. the second record (QC Name value is `Missing`) and the third record (AQI value is "`N/A`").

Kuala Lumpur	PM2.5 - Principal	2024-01-01 04:00 PM	2024	1	1	16	8.5	47	Good	9.0	UG/M3	1 Hr	Valid
Kuala Lumpur	PM2.5 - Principal	2024-01-01 05:00 PM	2024	1	1	17	8.5	47	Good	-999.0	UG/M3	1 Hr	Missing
Kuala Lumpur	PM2.5 - Principal	2024-01-01 07:00 PM	2024	1	1	19	-999.0	-999	N/A	16.0	UG/M3	1 Hr	Valid
Kuala Lumpur	PM2.5 - Principal	2024-01-01 08:00 PM	2024	1	1	20	13.5	59	Moderate	13.0	UG/M3	1 Hr	Valid

Rajah 3: Rekod sampel untuk dialih keluar.

Figure 3: Sample record to be removed.

Contoh Output (pada konsol):

Sample Output (on console):

```
Loading data from AQI sites...
Printing AQI sites summary...
+-------------------------------------------------------------------------+
| Site                | Year | #Record |  Valid | Invalid | Missing | Suspect |
+-------------------------------------------------------------------------+
| KualaLumpur         | 2024 |    5839 |   3359 |    2331 |     136 |      13 |
| Manila              | 2024 |    4612 |   4609 |       0 |       3 |       0 |
| Rangoon             | 2024 |    7111 |   7092 |      13 |       6 |       0 |
| Vientiane           | 2024 |    7057 |   7035 |       0 |      22 |       0 |
```

```
+--------------------------------------------------------------------------+
Generating AQI sites report...
Successfully written to KualaLumpur_2024.site
Successfully written to Manila_2024.site
Successfully written to Rangoon_2024.site
Successfully written to Vientiane_2024.site
```

Contoh Output (`KualaLumpur_2024.site` file):
Sample Output (`KualaLumpur_2024.site` fail):

```
Site : KualaLumpur
Year : 2024
AQI (Good) : 1375
AQI (Moderate) : 1832
AQI (Unhealthy for Sensitive Groups) : 44
AQI (Unhealthy) : 17
AQI (Very Unhealthy) : 9
AQI (Hazardous) : 8
Worst AQI : 696 (recorded 1 times)
Worst AQI Record : 2024-09-15 11:00 PM
Best AQI : 0 (recorded 23 times)
Best AQI Record : 2024-01-04 08:00 AM,2024-01-04 09:00 AM,2024-01-
04 10:00 AM,2024-01-08 05:00 AM,2024-01-08 03:00 PM,2024-01-16
01:00 PM,2024-01-16 02:00 PM,2024-01-16 03:00 PM,2024-01-16 06:00
PM,2024-01-17 12:00 AM,2024-01-17 03:00 AM,2024-01-17 04:00
AM,2024-01-25 02:00 PM,2024-01-25 03:00 PM,2024-01-25 04:00
PM,2024-01-31 05:00 AM,2024-01-31 06:00 AM,2024-03-30 03:00
PM,2024-04-16 06:00 AM,2024-04-16 07:00 AM,2024-04-16 08:00
AM,2024-04-17 12:00 PM,2024-10-18 01:00 PM
```

Kaedah `main` untuk program diberikan seperti di bawah (**MESTI IKUT**):
*`main` method for the program is given as below (**MUST FOLLOW**):*

```
public static void main(String[] args) {
    System.out.println("Loading data from AQI sites...");
    Mon_Site[] sites = readDataFromDirectory("path-to-the-folder");
    System.out.println("Printing AQI sites summary...");
    printSites(sites);
    System.out.println("Generating AQI sites report...");
    generateStatistics(sites);
}
```

(10 markah/*marks*)

TAMAT
END