
UNIVERSITI SAINS MALAYSIA

First Semester Examination
Academic Session 2016/2017

December 2016 / January 2017

EEE 430/4 – SOFTWARE ENGINEERING**[KEJURUTERAAN PERISIAN]**

Duration : 3 hours
[Masa : 3 jam]

Please check that this examination paper consists of **ELEVEN (11)** pages of printed material before you begin the examination. This examination paper consist of two versions, The English version and Malay version. The English version from page **TWO (2)** to page **SIX (6)** and Malay version from page **SEVEN (7)** to page **ELEVEN (11)**.

*Sila pastikan bahawa kertas peperiksaan ini mengandungi **SEBELAS (11)** muka surat bercetak sebelum anda memulakan peperiksaan ini. Kertas peperiksaan ini mengandungi dua versi, versi Bahasa Inggeris dan Bahasa Melayu. Versi Bahasa Inggeris daripada muka surat **DUA (2)** sehingga muka surat **ENAM (6)** dan versi Bahasa Melayu daripada muka surat **TUJUH (7)** sehingga muka surat **SEBELAS (11)**.*

Instructions: This question paper consists **FIVE (5)** questions. Answer **ALL** questions. All questions carry the same marks.

*[Arahan: Kertas soalan ini mengandungi **LIMA (5)** soalan. Jawab **SEMUA** soalan. Semua soalan membawa jumlah markah yang sama]*

Answer to any question must start on a new page

[Mulakan jawapan anda untuk setiap soalan pada muka surat yang baharu].

“In the event of any discrepancies, the English version shall be used”.

[Sekiranya terdapat sebarang percanggahan pada soalan peperiksaan, versi Bahasa Inggeris hendaklah diguna pakai].

ENGLISH VERSION

1. (a) List out the development activities in object-oriented software engineering and briefly explain each of them.

(30 marks)

- (b) What is the difference between a 'task' and an 'activity'? Provide an example in your answer.

(20 marks)

- (c) Based on UML, draw a class diagram representing a book defined by the following statement. Focus only on classes and relationships.

"A book is composed of a number of parts, which in turn are composed of a number of chapters. Chapters are composed of sections."

(10 marks)

- (d) From the answer in (c) above, extend the class diagram with its relationship to include the following attributes:

- (i) A book includes a publisher, publication date, and an ISBN
- (ii) A part includes a title and a number
- (iii) A chapter includes a title, a number, and an abstract
- (iv) A section includes a title and a number

(40 marks)

2. Consider a digital watch as a system and set the time 2 minutes behind.

- (a) Write down each interaction between the user and watch as a scenario. Record all interactions, including any feedback the watch provides.

(20 marks)

- (b) Identify the actor of the scenario and explain why these are the actors.

(5 marks)

- (c) Write the use case `SetTime` for this system. Include all possible cases, and include setting the time forward, backward, setting hours, minutes and seconds.
(25 marks)
- (d) Assuming that the system supports an alarm feature, describe setting the alarm time as a self-contained use case name `SetAlarmTime`.
(20 marks)
- (e) Examine the use cases in (c) and (d) above. Eliminate any redundancy and introducing using an 'include' relationship. Justify why an 'include' relationship is preferable to an extend relationship in this case.
(30 marks)
3. (a) During 'Analysis' activity, what UML diagrams are used here? What can a software engineer understand from this activity?
(15 marks)
- (b) Consider a file system with a graphical user interface such as Microsoft Windows Explorer. The following objects were identified from a use case describing how to copy a file from a pen-drive to a hard disk: `File`, `Icon`, `Trash`, `Folder`, `Disk`, `Pointer`. Take note that `Pointer` here is referring to the `Mouse`.
- (i) Specify which are entity objects, boundary objects and control objects. Explain.
(30 marks)
- (ii) Assuming the same file as in (b)(i) above, consider a scenario consisting of selecting a file on a pen-drive, dragging it to `Folder` and releasing the `Mouse`. Identify and define at least one control object associated with this scenario.
(15 marks)
- (iii) Arrange the objects listed from (b)(i) and (b)(ii) above horizontally on a sequence diagram. Draw the sequence of interactions resulting from dropping the file into a folder. Ignore exceptional cases.
(40 marks)

4. (a) Consider the following object model for Gregorian calendar (usual calendar system with months: January, February, March - December with possible number of days of 28, 29, 30, or 31 per month) Figure 4(a):

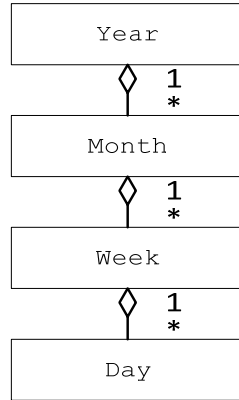


Figure 4(a)

- (i) Given your knowledge of the Gregorian calendar, list all the problems with this model. Modify the figure to correct each of them.

(20 marks)

- (ii) By using association multiplicity only, modify the model so that any developer who is unfamiliar with the Gregorian calendar could deduce the number of days in each month. Identify additional classes if necessary.

(30 marks)

- (b) Figure 4(b) below is a case model for a train ticket Distributor, where the common case functionality is modeled by PurchaseTicket use case and describing the steps necessary for a Passenger to successfully purchase a ticket.

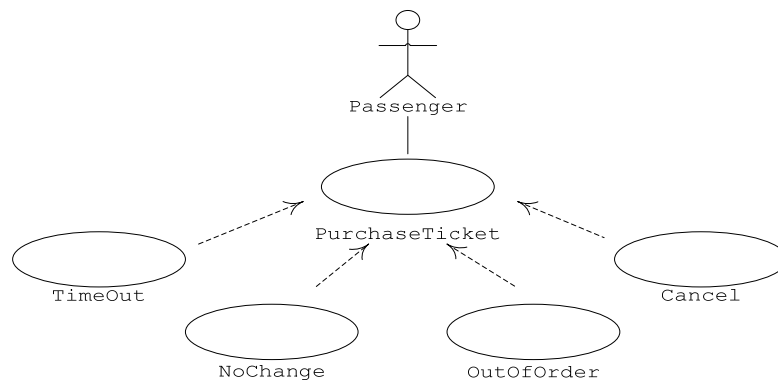


Figure 4(b)

The `TimeOut`, `NoChange`, `OutOfOrder` and `Cancel` use cases describe various exceptional conditions resulting from the state of the `Distributor` or action by the `Passenger`.

- (i) Come out with a use case to describe the normal interaction between the `Passenger` and the `Distributor`.

(40 marks)

- (ii) What are the three possible features of the `Distributor` that are likely to fail and should be tested?

(10 marks)

- 5. (a) You are designing the access control policies for a Web-based retail store. Customers access the store via the Web, browse product information, input their address and payment information, and purchase products. Suppliers can add new products, update product information, and receive orders. The store owner sets the retail prices, makes tailored offers to customers based on their purchasing profiles, and provides marketing services. You have to deal with three actors: 1) `StoreAdministrator`, 2) `Supplier`, and 3) `Customer`. Design an access control policy for all the three actors. `Customer(s)` can be created via the Web, whereas `Supplier(s)` are created by the `StoreAdministrator`. You can ignore other cases such as unregistered customers.

(50 marks)

- (b) Correct faults in the `isLeapYear()` and `getNumDaysInMonth()` from the following code and generate test cases using the path testing methods. You have to rewrite the whole code with the corrected syntaxes.

```
public class MonthOutOfBounds extends Exception {...};
public class YearOutOfBounds extends Exception {...};
class MyGregorianCalendar
{
    public static boolean isLeapYear(int year)
    {
        boolean leap;
        if ((year%4) == 0)
        {
            leap = true;
        }
        else
        {
            leap = false;
        }
        return leap;
    }
    public static int getNumDaysInMonth(int month, int year)
        throws MonthOutOfBounds, YearOutOfBounds
    {
        int numDays;
        if (year < 1)
        {
            throw new YearOutOfBounds(year);
        }
        if (month == 1 || month == 3 || month == 5 || month == 7
||month == 10 || month == 12)
        {
            numDays = 32;
        }
        else if (month == 4 || month == 6 || month == 9 || month ==
11)
        {
            numDays = 30;
        } else if (month == 2)
        {
            if (isLeapYear(year))
            {
                numDays = 29;
            } else
            {
                numDays = 28;
            }
        } else
        {
            throw new MonthOutOfBounds(month);
        }
        return numDays;
    }
}
```

(50 marks)

-oooOOooo-

...7/-

VERSI BAHASA MELAYU

1. (a) *Senaraikan aktiviti-aktiviti pembangunan dalam bidang kejuruteraan perisian berorientasikan objek dan terangkan secara ringkas setiap daripada mereka.*

(30 markah)

- (b) *Apakah perbezaan di antara 'tugas' dan 'aktiviti'? Berikan contoh dalam jawapan anda.*

(20 markah)

- (c) *Berdasarkan UML, lukiskan rajah kelas yang mewakili sebuah buku yang ditakrifkan oleh kenyataan berikut. Fokus hanya pada kelas dan hubungan.*

(10 markah)

“Satu buku terdiri daripada beberapa bahagian, yang seterusnya adalah terdiri daripada beberapa bab. Bab terdiri daripada bahagian.”

- (d) *Daripada jawapan (c) di atas, kembangkan rajah kelas tersebut dengan hubungkaitnya untuk memasukkan ciri-ciri berikut:*

(i) Buku termasuk penerbit, tarikh penerbitan, dan ISBN.

(ii) Bahagian termasuk tajuk dan bilangan

(iii) Bab termasuk tajuk, nombor, dan abstrak

(iv) Bahagian mengandungi tajuk dan bilangan

(40 markah)

2. *Pertimbangkan sebuah jam tangan digital sebagai satu sistem dan tetapkan masa 2 minit ke belakang.*

- (a) *Tuliskan setiap interaksi antara pengguna dan jam tangan sebagai senario. Rekodkan semua interaksi, termasuk apa-apa maklum balas yang jam tangan berikan.*

(20 markah)

- (b) *Kenal pasti pelakon senario dan jelaskan mengapa ini adalah pelakon.*

(5 markah)

- (c) *Tuliskan 'SetTime' kes guna untuk sistem ini. Masukkan semua kes yang mungkin termasuk menetapkan masa hadapan, ke belakang, menetapkan jam, minit dan saat.*

(25 markah)

(d) Dengan mengandaikan bahawa sistem ada ciri penggera, terangkan penetapan masa penggera sebagai kes guna serba lengkap 'SetAlarmTime'.

(20 markah)

(e) Kaji kes-kes guna dalam (c) dan (d) di atas. Hapuskan apa-apa lebih dan perkenalkan satu kes guna bahan 'specifyTime' dengan menggunakan hubungan 'termasuk'. Jelaskan mengapa hubungan 'termasuk' adalah bersesuaian untuk satu hubungan lanjutan dalam kes ini.

(30 markah)

3. (a) Semasa aktiviti 'Analisis', apakah gambarajah-gambarajah UML yang digunakan di sini? Apa yang boleh seorang jurutera perisian faham dari aktiviti ini?

(15 markah)

(b) Pertimbangkan sistem fail dengan antara muka pengguna grafik seperti Microsoft Windows Explorer. Objek berikut telah dikenal pasti dari kes penggunaan yang menerangkan cara untuk menyalin fail dari pendrive ke cakera keras: *File*, *Icon*, *Trash*, *Folder*, *Disk*, *Pointer*. Ambil perhatian bahawa *Pointer* di sini merujuk kepada *Mouse*.

(i) Tentukan objek entiti, objek sempadan dan objek kawalan. Jelaskan.

(30 markah)

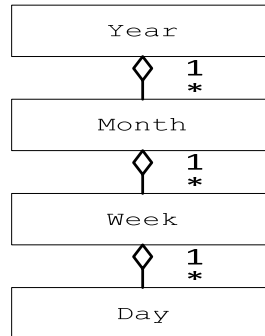
(ii) Dengan mengandaikan fail yang sama seperti dalam (b.) (i) di atas, pertimbangkan senario yang terdiri daripada memilih fail pada pendrive, menyeretnya ke *Folder* dan melepaskan *Mouse*. Kenalpasti dan tentukan sekurang-kurangnya satu objek kawalan yang berkaitan dengan senario ini.

(15 markah)

(iii) Susun objek yang disenaraikan dari (b)(i) dan (b)(ii) atas mendatar di atas rajah jujukan. Lukiskan urutan interaksi yang terhasil daripada menjatuhkan fail ke dalam folder. Abaikan kes yang luar biasa.

(40 markah)

4. (a) *Pertimbangkan model objek berikut untuk kalendar Gregorian (sistem kalendar biasa dengan bulan: Januari, Februari, Mac – Disember dengan jumlah kemungkinan hari sebanyak 28, 29, 30 ataupun 31 setiap bulan) Rajah 4(a):*



Rajah 4(a)

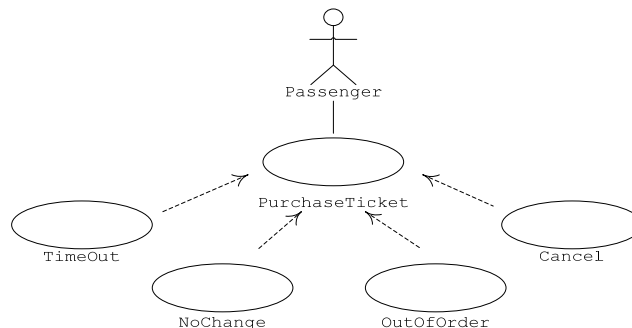
- (i) *Berdasarkan pengetahuan anda mengenai kalendar Gregorian, senaraikan semua masalah dengan model objek di atas. Ubahsuai model tersebut untuk membetulkannya.*

(20 markah)

- (ii) *Dengan menggunakan kepelbagaian penyekutuan, ubahsuai model tersebut supaya sebarang pembangun sistem yang tidak biasa dengan kalendar Gregorian dapat mendeduksi bilangan hari dalam setiap bulan. Kenalpasti kelas tambahan, sekiranya perlu.*

(30 markah)

- (b) *Rajah 4(b) di bawah adalah satu model kes untuk sistem pembelian tiket train, Distributor, di mana fungsi kes biasa dimodelkan dengan PurchaseTicket kes guna untuk menerangkan langkah-langkah yang perlu untuk seorang penumpang (Passenger) untuk membeli satu tiket.*



Rajah 4(b)

Kes guna dalam rajah di atas iaitu TimeOut, NoChange, OutOfOrder dan Cancel adalah untuk keadaan yang luar biasa hasil daripada keadaan Distributor ataupun tindakan Passenger.

(i) Hasilkan satu kes guna untuk menerangkan interaksi normal di antara Passenger dan Distributor.

(40 markah)

(ii) Berikan tiga ciri-ciri Distributor yang berkemungkinan untuk gagal dan patut diuji?

(10 markah)

5. (a) *Anda sedang mereka bentuk dasar kawalan akses untuk satu perniagaan berasaskan web. Pelanggan-pelanggan (customers) mengakses laman web, melayari maklumat produk, memasukkan alamat mereka dan juga maklumat pembayaran, dan akhirnya membeli produk. Pembekal-pembekal (suppliers) boleh menambah produk baru, mengemaskini maklumat produk dan juga menerima pesanan. Pemilik perniagaan boleh meletakkan harga runcit produk, menyediakan tawaran yang bersesuaian dengan profil pembelian pelanggan, dan jugak menyediakan servis pemasaran. Anda dikehendaki berurusan dengan tiga pelakon: 1) StoreAdministrator, 2) Supplier, dan 3) Customer. Rekabentuk satu polisi kawalan akses untuk ketiga-tiga pelakon tersebut. Customer boleh diwujudkan/didaftarkan melalui laman web, manakala Supplier didaftarkan oleh StoreAdministrator. Anda boleh abaikan kes-kes lain seperti Customer yang tidak berdaftar.*

(50 markah)

(b) *Betulkan kesalahan di dalam isLeapYear() dan getNumDaysInMonth() daripada kod berikut dan hasilkan kes ujian dengan menggunakan kaedah-kaedah ujian laluan. Anda dikehendaki menulis keseluruhan kod bersekali dengan sintaks-sintaks yang telah diperbetul.*

```
public class MonthOutOfBounds extends Exception {...};
public class YearOutOfBounds extends Exception {...};

class MyGregorianCalendar
{
    public static boolean isLeapYear(int year)
    {
        boolean leap;
        if ((year%4) == 0)
        {
            leap = true;
        }
        else
        {
            leap = false;
        }
        return leap;
    }
    public static int getNumDaysInMonth(int month, int year)
        throws MonthOutOfBounds, YearOutOfBounds
    {
        int numDays;
        if (year < 1)
        {
            throw new YearOutOfBounds(year);
        }
        if (month == 1 || month == 3 || month == 5 || month == 7
||month == 10 || month == 12)
        {
            numDays = 32;
        }
        else if (month == 4 || month == 6 || month == 9 || month ==
11)
        {
            numDays = 30;
        } else if (month == 2)
        {
            if (isLeapYear(year))
            {
                numDays = 29;
            } else
            {
                numDays = 28;
            }
        } else
        {
            throw new MonthOutOfBounds(month);
        }
        return numDays;
    }
}
```

(50 markah)