UNIVERSITI SAINS MALAYSIA

First Semester Examination
2014/2015 Academic Session

December 2014/January 2015

CPT113 – Programming Methodology & Data Structures
[Methodologi Pengaturcaraan & Struktur Data]

Duration : 2 hours
[Masa : 2 jam]

INSTRUCTIONS TO CANDIDATE:
[ARAHAN KEPADA CALON:]

- Please ensure that this examination paper contains **FOUR** questions in **THIRTHEEN** printed pages before you begin the examination.

  *[Sila pastikan bahawa kertas peperiksaan ini mengandungi **EMPAT** soalan di dalam **TIGA BELAS** muka surat yang bercetak sebelum anda memulakan peperiksaan ini.]*

- Answer **ALL** questions.

  *[Jawab **SEMUA** soalan.]*

- You may answer the questions either in English or in Bahasa Malaysia.

  [Anda dibenarkan menjawab soalan sama ada dalam bahasa Inggeris atau bahasa Malaysia.]

- 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.]

1. (a) Given the following C++ program:

```
class Base
{
public:

   void func();
   void print() const;
   Base();
   Base(int, double)

private:
   int food,;
   double leisure;
};
Base expenses;
```

(i) How many constructors does `class Base` have?

(ii) Write the definition of the member function `func` so that `food` is set to 10, `leisure` is set to15.5.

(iii) Write the definition of the default constructor of the `class Base` so that the private data members are initializes to 0.

(iv) Write the definition of the member function `print` that prints the content of `food` and `leisure`.

(vi) Write a C++ statement that prints the value of data members of the object `expenses`.

(15/100)

(b) What is the output of the following program?

```
#include <iostream>
using namespace std;
class baseClass
{
public:
   void print () const;
   int getX();
   baseClass (int a=0);

protected:
   int x;
};
```

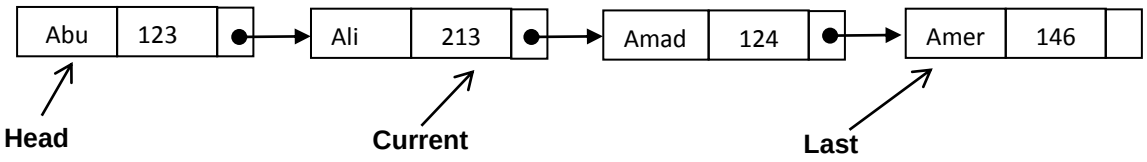
```
class derivedClass: public baseClass
{
public:
void print () const;
int getResult();
derivedClass (int a=0, int b=0);

protected:
   int y;
};

void baseClass::print () const
{
   cout << "In base: x=" <<x<<endl;
}

baseClass:: baseClass( int a)
{
   x=a;
}

int baseClass :: getX()
{
   return x;
}

void derivedClass:: print() const
{
   cout<<"In derived: x= " << x<<", y = "<<y<<",
         x+y="<< x+y<<endl;
}

int derivedClass::getResult()
{
   return x+y;
}

derivedClass::derivedClass (int a, int b): baseClass(a)
{
   y=b;
}

int main()
{
   baseClass baseObject(7);
   derivedClass derivedObject(3,8);

   baseObject.print();
   derivedObject.print();

   cout<<"**** "<<baseObject.getX()<< endl;
   cout<<"###"<<derivedObject.getResult()<< endl;
   return 0;
}
```

2. (a) What is the output of the following program?

```
int x;
int *p;
int *q;
p = new int[10];
q = p;
*p = 4;

for (int j = 0; j < 10;j++)
{
   x = *p;
   p++;
   *p = x+j;
}

for (int k = 0; k < 10;k++)
{
   cout <<*q<< " ";
   q++;
}

cout<<endl;
```

(5/100)

(b) Given the following linked list that stores information about the name and matrix number (ID) of the students in the School of Computer Sciences.



Write C++ statements to do the following on the linked list above:

(i) Use `struc` to define the above nodes which consists of variable `names` and `ID`.

(ii) Declare the pointers `Head`, `Current` and `Last`.

(iii) Delete the node content `name` Amad and `ID` 124.

(iv) Create and insert the node with `name` Alia and `ID` 134 after node with `name` Ali and `ID` 213.

(12/100)

(c) Based on the linked list in Question 2(b), write a function `search` using C++ statements to search for a node with `name` Amer and `ID` 146.

Note: you must use a `while` loop

(8/100)

3. (a) Given the following recursive function:

```
int (fibNum(int a, int b, int n))
{
   if (n==1)
      return a;
   else if (n==2)
      return b;
   else
      return fibNum(a, b, n – 1) + fibNum(a, b, n – 2)
```

(i) Trace the execution of the following statement step-by-step to derive your final answer.

```
cout << fibNum (2, 3, 5) << endl;
```

(ii) State the base case(s).

(iii) State the general case(s).

(9/100)

(b) Draw a Binary Tree for the following keys:
9, 7, 14, 4, 5, 11, 21, 12, 13, 10.

(4/100)

(c) Based on the Binary Tree in Question 3(b), list the sequence or path of the following Binary Tree traversals.

(i) Pre-order traversal.

(ii) Post-order traversal.

(iii) In-order traversal.

(9/100)

4. (a) Study the following stack and queue class definition implemented using linked list:

```
template <class Type>
struct nodeType
{
   Type info;
   nodeType<Type> *link;
};

template<class Type>
class linkedstackType
{
public:
   const linkedstackType<Type>& operator=(const
      linkedstackType<Type>&);
   void initializeStack();
   bool isEmptyStack();
   bool isFullStack();
   void push(const Type& newItem);
   void pop(Type& poppedElement);
   void destroyStack();
   linkedstackType();
   linkedstackType(const linkedstackType<Type>& otherStack);
   ~linkedstackType();
private:
   nodeType<Type> *stacktop;
};

template<class Type>
class linkedqueueType
{
public:
   const linkedqueueType<Type>& operator= (const
      linkedqueueType<Type>&);
   bool isEmptyQueue();
   bool isFullQueue();
   void destroyQueue();
   void initializeQueue();
   void addQueue(const Type& newElement);
   void deleteQueue(Type& deqElement);
   linkedqueueType ();
   linkedqueueType(const linkedqueueType<Type>& otherQueue);
   ~linkedqueueType();

private:
   nodeType<Type> *front;
   nodeType<Type> *rear;
};
```

(i) What are the differences between implementations of `Stack` and `Queue`? Justify your answers.

(4/100)

(ii) Write the C++ statements for the `addQueue` function.

(6/100)

(iii) Write the C++ statements for the `deleteQueue` function.

(6/100)

(b) You are provided with the following information:

- Test marks for CPT113 are listed as follows:

Name	**Marks**
Ahmad	80
Siti	77
Yang	78
Geobin	75
Henry	70

- There are some changes on the given marks:

  Siti's marks has been increased from 77 to 78
  Henry's mark has been increased from 70 to 74
  Ahmad's mark has been reduced from 80 to 79

Write a complete `main` function using the abstract class definition from 4(a) to perform the following tasks (assume that the functions to implement the queue operations are included in the header file "**linkedQueue.h**"):

- Input the student's name and student's marks into the queue accordingly.
- Display their names and marks accordingly.

(12/100)

1. (a) Diberi atur cara C++ berikut:

```
class Base
{
public:

   void func();
   void print() const;
   Base();
   Base(int, double)

private:
   int food,;
   double leisure;
};
Base expenses;
```

(i) Berapakah bilangan pembina yang dipunyai oleh `Class Base`?

(ii) Tulis takrifan bagi fungsi ahli `func` supaya `food` ditetapkan kepada 10, dan `leisure` ditetapkan kepada 15.5.

(iii) Tulis takrifan bagi pembina lalai untuk `Class Base` supaya ahli-ahli data sulit di umpuk kepada 0.

(iv) Tulis takrifan bagi fungsi ahli `print` yang mencetak kandungan bagi `food` dan `leisure`.

(v) Tulis kenyataan-kenyataan C++ bagi mencetak nilai data ahli bagi objek `expenses`.

(15/100)

(b) Apakah output bagi atur cara berikut?

```
#include <iostream>
using namespace std;
class baseClass
{
public:
   void print () const;
   int getX();
   baseClass (int a=0);

protected:
   int x;
};
```

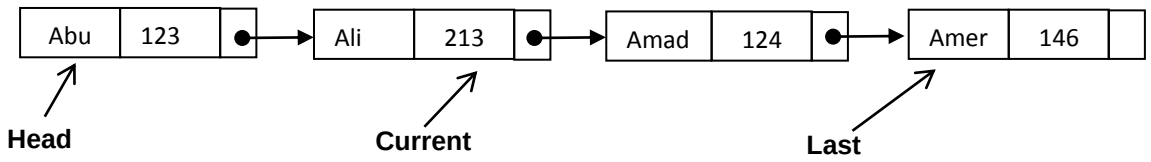
```
class derivedClass: public baseClass
{
public:
void print () const;
int getResult();
derivedClass (int a=0, int b=0);

protected:
   int y;
};

void baseClass::print () const
{
   cout << "In base: x=" <<x<<endl;
}

baseClass:: baseClass( int a)
{
   x=a;
}

int baseClass :: getX()
{
   return x;
}

void derivedClass:: print() const
{
   cout<<"In derived: x= " << x<<", y = "<<y<<",
         x+y="<< x+y<<endl;
}

int derivedClass::getResult()
{
   return x+y;
}

derivedClass::derivedClass (int a, int b): baseClass(a)
{
   y=b;
}

int main()
{
   baseClass baseObject(7);
   derivedClass derivedObject(3,8);

   baseObject.print();
   derivedObject.print();

   cout<<"**** "<<baseObject.getX()<< endl;
   cout<<"###"<<derivedObject.getResult()<< endl;
   return 0;
}
```

2. (a) Apakah output bagi atur cara berikut?

```
int x;
int *p;
int *q;
p = new int[10];
q = p;
*p = 4;

for (int j = 0; j < 10;j++)
{
   x = *p;
   p++;
   *p = x+j;
}

for (int k = 0; k < 10;k++)
{
   cout <<*q<< " ";
   q++;
}

cout<<endl;
```

(5/100)

(b) Diberi senarai berpaut berikut yang menyimpan maklumat berkenaan nama dan nombor matrik (ID) bagi pelajar Pusat Pengajian Sains.



Tulis kenyataan-kenyataan C++ untuk melaksanakan berikut ke atas senarai berpaut di atas:

(i) Guna `struc` untuk menakrifkan nod di atas yang mengandungi pemboleh ubah `nama` dan ID.

(ii) Isytihar penuding-penuding `Head`, `Current` dan `Last`.

(iii) Hapus nod yang mengandungi `nama` Amad dan `ID` 124.

(iv) Cipta dan selit nod bagi `nama` Alia and `ID` 134 selepas nod bagi `nama` Ali dan `ID` 213.

(12/100)

(c) Berdasarkan senarai berpaut di Soalan 2(b), tulis fungsi `search` menggunakan kenyataan C++ untuk mencari nod bagi `nama` Amer and `ID` 146.

Nota: Anda mesti menggunakan gelung `while`.

(8/100)

3. (a) Diberi fungsi rekursif yang berikut:

```
int (fibNum(int a, int b, int n))
{
   if (n==1)
      return a;
   else if (n==2)
      return b;
   else
      return fibNum(a, b, n – 1) + fibNum(a, b, n – 2)
```

(i) Jejak pelaksanaan kenyataan yang berikut langkah demi langkah untuk mendapat jawapan akhir anda.

```
cout << fibNum (2, 3, 5) << endl;
```

(ii) Nyatakan kes-kes asas.

(iii) Nyatakan kes-kes umum.

(9/100)

(b) Lukis pepohon perduaan bagi kekunci yang berikut:
9, 7, 14, 4, 5, 11, 21, 12, 13, 10

(4/100)

(c) Berdasarkan pepohon perduaan dari Soalan 3(a), senaraikan jujukan atau laluan bagi bagi penyusuran pepohon perduaan yang berikut:

(i) Penyusuran tertib Awalan.

(ii) Penyusuran tertib Akhiran.

(iii) Penyusuran tertib Sisipan.

(9/100)

4. (a) Kaji takrifan kelas tindanan dan baris gilir yang diimplementasikan menggunakan senarai berpaut berikut:

```
template <class Type>
struct nodeType
{
   Type info;
   nodeType<Type> *link;
};

template<class Type>
class linkedstackType
{
public:
   const linkedstackType<Type>& operator=(const
      linkedstackType<Type>&);
   void initializeStack();
   bool isEmptyStack();
   bool isFullStack();
   void push(const Type& newItem);
   void pop(Type& poppedElement);
   void destroyStack();
   linkedstackType();
   linkedstackType(const linkedstackType<Type>& otherStack);
   ~linkedstackType();
private:
   nodeType<Type> *stacktop;
};

template<class Type>
class linkedqueueType
{
public:
   const linkedqueueType<Type>& operator= (const
      linkedqueueType<Type>&);
   bool isEmptyQueue();
   bool isFullQueue();
   void destroyQueue();
   void initializeQueue();
   void addQueue(const Type& newElement);
   void deleteQueue(Type& deqElement);
   linkedqueueType ();
   linkedqueueType(const linkedqueueType<Type>& otherQueue);
   ~linkedqueueType();

private:
   nodeType<Type> *front;
   nodeType<Type> *rear;
};
```

(i) Apakah perbezaan antara pelaksanaan tindanan dan baris gilir. Justifikasikan jawapan anda.

(4/100)

(ii) Tulis kenyataan-kenyataan C++ untuk fungsi `addQueue`.

(6/100)

(iii) Tulis kenyataan-kenyataan C++ untuk fungsi `deleteQueue`.

(6/100)

(b) Anda disediakan dengan maklumat yang berikut:

- Markah ujian CPT113 adalah disenaraikan seperti yang berikut:

Nama	**Markah**
Ahmad	80
Siti	77
Yang	78
Geobin	75
Henry	70

- Terdapat sedikit perubahan dalam pemarkahan yang telah diberikan:

  Markah Siti telah meningkat dari 77 ke 78
  Markah Henry telah meningkat dari 70 ke 74
  Markah Ahmad telah berkurang dari 80 ke 79

Tulis fungsi `main` yang lengkap menggunakan takrifan kelas abstrak daripada 4(a) untuk melaksanakan tugas-tugas berikut (andaikan bahawa fungsi untuk pelaksanaan operasi baris gilir telah terkandung dalam kepala fail "**linkedQueue.h**"):

- Masukkan nama dan markah pelajar ke dalam baris gilir mengikut yang sepatutnya.
- Paparkan nama dan markah mengikut yang sepatutnya.

(12/100)