UNIVERSITI SAINS MALAYSIA

First Semester Examination
2015/2016 Academic Session

December 2015/January 2016

CPT113 – Programming Methodology & Data Structures
[Metodologi 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 **ELEVEN** printed pages before you begin the examination.

  *[Sila pastikan bahawa kertas peperiksaan ini mengandungi **EMPAT** soalan di dalam **SEBELAS** 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 the class `Base` have?

(ii) Write the definition of the member function `func` so that `food` is set to 10, `leisure` is set 15.5.

(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`.

(v) Write a C++ statement(s) 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;
};
```

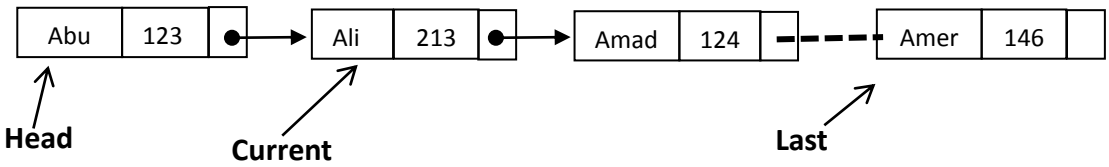
```
class derivedClass: public baseClass
{
public:
   void print () const;
   int getResult();
   derivedClass (int a =0, int b =0);

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 matric number (`ID`) of the students in School of Computer Sciences.



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

(i) Use `struc` to define the above node 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++ statement to search for a node with `name` Amer and `ID` 146.
Note: You must use `while` loop.

(8/100)

3. (a) Evaluate the following postfix notation of expression (Show status of stack after execution of each operation):

32, 4, /,2,*,12,3, -,+

(6/100)

(b) Given is the abstract class definition as an ADT:

```
Template <class Type>
class stackType {
   public:
      const StackType<Type> & operator= (const listType<Type>&);
      void initializeStack();
      bool isEmptyStack();
      bool isFullStack();
      void destroyStack();
      void push(const Type& newItem);
      void top();
      void pop();
      stackType(int stacksize = 100);
      ~ stackType();

   private:
      int maxStackSize;
      int stackTop;
   Type *list;

};
```

(i) Write C++ statements for the `initializeStack` function.

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

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

(iv) Write C++ statements for the `pop` function.

(v) Write C++ statements for the `destroyStack` function.

(15/100)

(c) Convert the above abstract class ADT definition in question 3(b) using Linked List.

(5/100)

4. (a) Assume the following `struct` definition of `NodeType` as follows:

```
struct NodeType
{
   elemenType info;
   NodeType *left_link;
   NodeType *right_link;
};
```

Assume the pointer `P` of `NodeType` is a current pointer

(i) Write C++ statements for function postorder traversal of the binary tree.

(ii) Write C++ statements for function inorder traversal of the binary tree.

(iii) Write C++ statements for function preoder traversal of the binary tree.

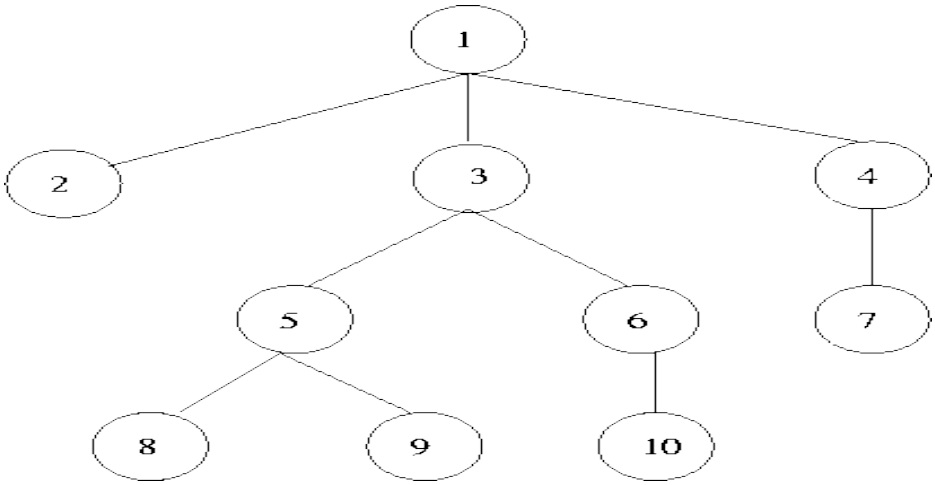
(12/100)

(b) Based on the below Binary Tree, traverse the tree using

(i) Preorder

(ii) Postorder

(iii) Inorder



(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-pembina class `Base`?

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

(iii) Tulis definisi bagi pembina lalai untuk class `Base` supaya ahli-ahli data sulit diasalkan kepada 0.

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

(v) Tulis 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;
};
```

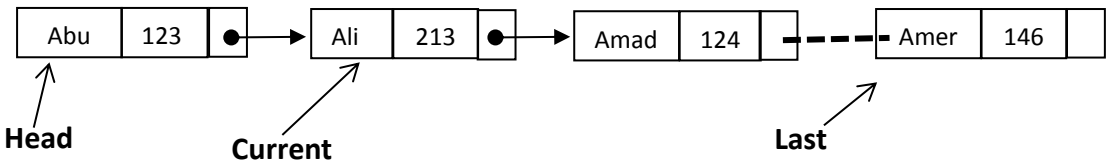
```
class derivedClass: public baseClass
{
public:
   void print () const;
   int getResult();
   derivedClass (int a =0, int b =0);

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 Komputer.



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 bagi `nama` Amad dan `ID` 124.

(iv) Bina and selit nod bagi `nama` Alia dan `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 dan `ID` 146.
Nota: Anda mesti menggunakan gelung `while`.

(8/100)

3. (a) Nilai ungkapan notasi postfix berikut. (Tunjuk status tindanan selepas pelaksanaan setiap operasi):

32, 4, /,2,*,12,3, -,+

(6/100)

(b) Diberi definisi kelas abstrak sebagai ADT:

```
Template <class Type>
class stackType {
   public:
      const StackType<Type> & operator= (const listType<Type>&);
      void initializeStack();
      bool isEmptyStack();
      bool isFullStack();
      void destroyStack();
      void push(const Type& newItem);
      void top();
      void pop();
      stackType(int stacksize = 100);
      ~ stackType();

   private:
      int maxStackSize;
      int stackTop;;
   Type *list;

};
```

(i) Tulis kenyataan C++ untuk fungsi `initializeStack`.

(ii) Tulis kenyataan C++ untuk fungsi `push`.

(iii) Tulis kenyataan C++ untuk fungsi `isFullStack`.

(iv) Tulis kenyataan C++ untuk fungsi `pop`.

(v) Tulis kenyataan C++ untuk fungsi `destroyStack`.

(15/100)

(c) Tukar definisi ADT kelas abstrak di soalan 3(b) dengan menggunakan Senarai Berpaut.

(5/100)

4. (a) Andaikan definisi `struct NodeType` seperti di bawah.

```
struct NodeType
{
   elemenType info;
   NodeType *left_link;
   NodeType *right_link;
};
```

Andaikan penunding `P` jenis `NodeType` adalah penunjuk semasa

(i) Tulis kenyataan C++ untuk fungsi *postorder* bagi penyusuran pohon perduaan.

(ii) Tulis kenyataan C++ untuk fungsi *inorder* bagi penyusuran pohon perduaan.

(iii) Tulis kenyataan C++ untuk fungsi *preoder* bagi penyusuran pohon perduaan.

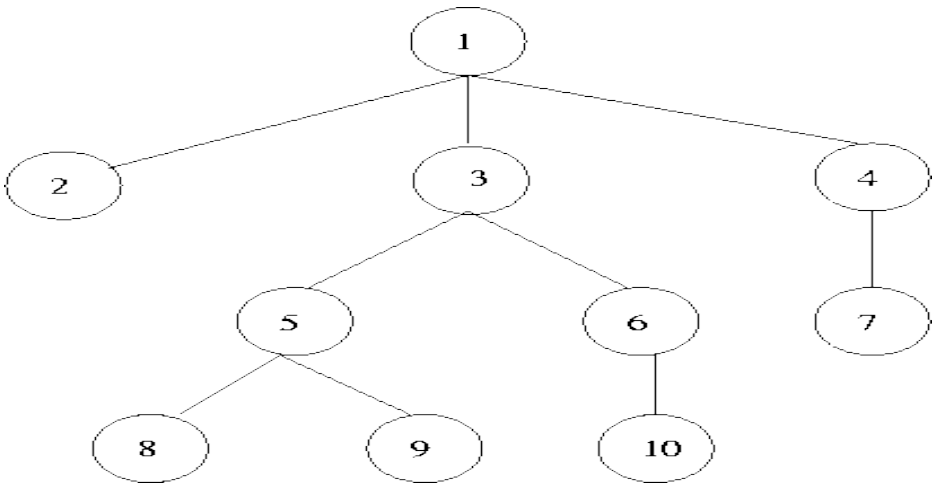
(12/100)

(b) Berdasarkan maklumat pohon perduaan di bawah, jelajah pohon menggunakan:

(i) *Preorder*

(ii) *Postorder*

(iii) *Inorder*



(12/100)