CONFIDENTIAL

Infrastructure University Kuala Lumpur

FINAL EXAMINATION
SEPTEMBER/OCTOBER SEMESTER 2018

ADVANCED PROGRAMMING
(CSC 3530)

(TIME : 3 HOURS)

MATRIC NO. : | | | | | | | | | |

IC. / PASSPORT NO. : | | | | | | | | | | | | | | |

LECTURER : **AMAR SUBEDI**

GENERAL INSTRUCTIONS

1. This question booklet consists of 9 printed pages including this page.
2. **SECTION A**: Answer **TWO (2) questions** in the **ANSWER BOOKLET**
3. **SECTION B**: Answer **ALL questions** in the **ANSWER BOOKLET**
4. **SECTION C**: Answer **ALL questions** in the **ANSWER BOOKLET**

CONFIDENTIAL

INSTRUCTIONS: **TIME: 3 *HOURS***

SECTION A **(40 MARKS)**

There are THREE (3) questions in this section. Answer TWO (2) Questions in the Answer Booklet.

1. Given the following code of `SimpleVector` class to allocate values to an array pointer. A constructor sets the size of the array and allocates memory for it. The `arraySize` member variable is declared as an `int`, regardless of the data type of the array.

   Rewrite the following code into a class template:

```
class SimpleVector {
private:
  int *aptr;        //point to the allocated array
  int arraySize;    //number of elements in the array
  void memError(); //Handles memory allocation errors
public:
  SimpleVector()  //Default constructor
  { aptr = 0; arraySize = 0; }

  SimpleVector(int);  //Constructor declaration
  SimpleVector(const SimpleVector &);
  ~SimpleVector();  //Destructor declaration
  int size() const { return arraySize; }
  int getElementAt(int position); //to return an element
  int &operator[] (const int &);  //Overloaded []operator
};

SimpleVector::SimpleVector(int s) {
    arraySize = s;
    try { aptr = new int[s]; }
    catch(bad_alloc) { memError(); }

    for(int count = 0; count < arraySize; count++)
        *(aptr + count) = 0;     }

SimpleVector::SimpleVector(const SimpleVector &obj) {
    arraySize = obj.arraySize;
    aptr = new int[arraySize];
    if(aptr == 0) memError(); //if Can't allocate memory

    for(int count = 0; count < arraySize; count++)
        *(aptr + count) = *(obj.aptr + count);     }
```

```
SimpleVector::~SimpleVector() {
    if(arraySize > 0) delete[] aptr;  }

void SimpleVector::memError() { exit(EXIT_FAILURE); }

int SimpleVector::getElementAt(int sub) {
    if (sub < 0 || sub >= arraySize) //Subscript out of range
        exit(EXIT_FAILURE);
   return aptr[sub]; }

int &SimpleVector::operator[](const int &sub) {
    if (sub < 0 || sub >= arraySize) //Subscript out of range
        exit(EXIT_FAILURE);
   return aptr[sub];  }
```

[CLO2: PLO2: C3](20 marks)

2. Consider the following code:

```
int doSomething(int &x, int &y)  {
    int temp = x;
    x = y * 10;
    y = temp * 10;
    return x + y;    }
```

Answer the following questions:

a) Given the function call for the above code:

```
int xx = 2, yy = 3;
doSomething(xx, yy);
```

i. What is the final value for xx, yy, x and y?

[CLO1: PLO1: C1](4 marks)

ii. What is the returning value from the `doSomething()` function?

[CLO1: PLO1: C1](1 mark)

b) Rewrite the function so it uses pointers instead of reference variables.

[CLO2: PLO2: C3](6 marks)

c) Write a function call to pass the arguments for answer in question 2(b).

[CLO2: PLO2: C3](3 marks)

d) Using a pointer notation, rewrite the `doSomething()` function to swap both argument values of x and y. The function shall display the updated values (no return value is needed).

[CLO2: PLO2: C3](6 marks)

3. Given the `StudentTotal` class to identify a total student for each faculty:

```
class StudentTotal {
private:
    int totalByFac; //instance member
public:
    StudentTotal(){ totalByFac = 0; }
    int getTotalByFac() { return totalByFac; }
};
```

Answer the following questions:

a) Declare a static member variable as to store the total number of all students.

[CLO2: PLO2: C3](2 marks)

b) Initialize a static member variable in question 3(a) with a 0.

[CLO2: PLO2: C3](2 marks)

c) In a `StudentTotal` class, write a void function to accept an argument of current student number. The function should update the student number, with the following pseudocode:

Total student by faculty += current student number
Total all student number += current student number

[CLO2: PLO2: C3](6 marks)

d) Consider the following main():

```
int main()
{
    int count; // Loop counter
    double totalStudent; //Variable for student's number
    const int NUM_FAC = 4; //Number of divisions
    StudentTotal byFac[NUM_FAC]; //Array of Budget objects
    // … continue writing code from here    }
```

i. Write code to ask user for total number for each faculty. By using the array object, pass the input value to a function created in question 3(c).

[CLO2: PLO2: C3](5 marks)

ii. By using the array object, write code to display total number for each faculty.

[CLO2: PLO2: C3](5 marks)

SECTION B **(30 MARKS)**

There are FIVE (5) questions in this section. Answer ALL questions in the Answer Booklet.

1. Consider the following code:

```
class B  {
   public:
      void f(int x)
      { cout << "\nfrom class B, function F(): " << x; }
      virtual void g(int x)
      { cout << "\nfrom class B, function G(): " << x; }
};
class D:public B  {
   public:
      void f(int x)
      { cout << "\nfrom class D, function F(): " << x; }
      void g(int x)
      { cout << "\nfrom class D, function G(): " << x; }
};
```

What is the output if:

a)
```
D d;
B *bp = &d;
bp -> f(3);
bp -> g(5);
```

[CLO1: PLO1: C1](2 Marks)

b)
```
D d;
D *dp = &d;
dp -> f(4);
dp -> g(6);
```

[CLO1: PLO1: C1](2 Marks)

2. What is the output by the following program segment?

```
int main() {
 Rectangle box1(10.0, 10.0); //width = 10.0, length = 10.0
 Rectangle box2(20.0, 20.0); //width = 20.0, length = 20.0

 cout << box1.getWidth() << "  " << box1.getLength() << endl;
 cout << box2.getWidth() << " " << box2.getLength() << endl;

 box2 = box1;
 cout << box1.getWidth() << "  " << box1.getLength() << endl;
 cout << box2.getWidth() << " " << box2.getLength() << endl;
return 0;  }
```

[CLO1: PLO1: C1](4 Marks)

3. Given code below has errors. Identify THREE (3) errors, state the line number involved and explain:

```
1   class Tank : public Cylinder {
2   private:
3      int fuelType;
4      double gallons;
5   public:
6      Tank();
7      ~Tank(double);
8      void setContents(double a);
9      void setContents(double);
10  };
```

[CLO2: PLO2: C3](9 Mark)

4. Given code below has errors. Identify THREE (3) errors, state the line number involved and explain:

```
1   int x = 5, y, z;
2   const int *const ptrA = &x;
3   int *const ptrB = &z;
4
5   *ptrA = 7;
6   PtrA = &y;
7   *ptrB = 7;
8   Ptr = &y;
```

[CLO2: PLO2: C3] (9 Mark)

5. Given the following code, assuming `arr` is an array of `int`, identity the output:

a)
```
if (arr < &arr[1])
   cout << "True";
else
   cout << "False";
```
[CLO1: PLO1: C1](1 Mark)

b)
```
if (&arr[4] < &arr[1])
   cout << "True";
else
   cout << "False";
```
[CLO1: PLO1: C1](1 Mark)

c)
```
if (arr != &arr[2])
   cout << "True";
else
   cout << "False";
```
[CLO1: PLO1: C1](1 Mark)

d)
```
if (arr != &arr[0])
   cout << "True";
else
   cout << "False";
```
[CLO1: PLO1: C1](1 Mark)

SECTION C **(30 MARKS)**

There is ONE (1) question in this section. Answer ALL question in the Answer Booklet.

1. The following declaration class `Date` store a date in three integers: month, day and year.

```
class Date
{
  private:
   int month;  // To hold the month
   int day;    // To hold the day
   int year;   // To hold the year

  public:
   Date(int, int, int);
   void setMonth(int);
   void setDay(int);
   void setYear(int);

   string monthNames(); //Functions to identify months in a string

   //Functions to print the date
   void showDate1();
   void showDate2();
   void showDate3();

   //Exception classes
   class InvalidDay{ };
   class InvalidMonth{ };
};
```

There should be member functions to print the date in the following forms:
12/25/2014
December 25, 2014
25 December 2014

The class should implement the following exception classes:

- `InvalidDay`: Throw when an invalid day (< 1 or > 31) is passed to the class.
- `InvalidMonth`: Throw when an invalid month (< 1 or > 12) is passed to the class.

Write a definition class `Date` (implementation file), which include the following function:

```
string monthNames()
{
    if (month == 1) return "January";
    else if(month == 2) return "February";
    else if(month == 3) return "March";
    else if(month == 4) return "April";
    else if(month == 5) return "May";
    else if(month == 6) return "June";
    else if(month == 7) return "July";
    else if(month == 8) return "August";
    else if(month == 9) return "September";
    else if(month == 10) return "October";
    else if(month == 11) return "November";
    else return "December";
}
```

[CLO3: PLO6: C5](30 Marks)

**** END OF QUESTIONS *****