National University of Computer and Emerging Sciences, Lahore Campus

Course:	Computer Programming	Course Code:	CS103
Program:	BS(Computer Science)	Semester:	Spring 2017
Duration:	60 Minutes	Total Marks:	30
Paper Date:	22-Feb-2017	Weight	15%
Section:	All	Page(s):	3
Exam:	Midterm-I	Roll No:	

Instruction/Notes: You can take extra sheets for rough work but not attach with this paper.

Question 1: **(20 marks)**

a) Consider the following program:

```
int main() {

   int x, y, z;
   int *p = &x, *q = &y, *r = &z;
   add(p, q, r);
   cout << "Sum of x and y: " << z;
   return 0;
}
```

Write code of "add".

b) Consider the following program:

```
int main() {

   int* p = new int[100];
   int* q = p;
   mid(q);
   *q = 234;
   cout << p[50]; //Output shall be 234
   delete [] p;
   return 0;
}
```

Write code of "mid".

c) Consider the following program:

```
int main() {

   int* p = new int[100];
   int* q = p;
   mid(&q);   // Note the & sign
   *q = 234;
   cout << p[50]; //Output shall be 234
   delete [] p;
   return 0;
}
```

Write code of "mid".

d) Correct any error(s) in the following program:

```
int* foo() {
      int* a[100];
      a[0] = new int [20];
      a[1] = new int [30];
      a[2] = new int [40];
      // ...
      return a;
}

int main() {
      int* p = foo();
      for (int i = 0; i < 100; ++i)
            delete [] p[i];
      return 0;
}
```

e)

Write only corrected lines of code.

Question 2 (Section C, D) **(10 marks)**

Write a C++ function to compute intersection of two given lists. Use the following definition and prototype:

```
struct Node {
      int x;
      Node* next;
};

Node* intersect(Node* p, Node* q); // p and q are not null
```

Here "p" points to the first list, and "q" to the second. The function shall develop a new list, and shall return a pointer to this new list.

You can use the following function to add members in the new list:

```
void add(Node* head, Node* tail, int y);
```

Here "y" holds the integer that we want to add into a list pointed by "head" and "tail". You do not need to write code for this function.

Question 2 (Section A, B, E, F, R) **(10 marks)**

Write a function **void RemoveRepetition (int** Arr, int rows)** that takes number of rows and a two-dimensional array ***Arr,*** which containspositive integers as input. Number of columns may vary in each row, so the last element of each row contains value (-1) to indicate the end of row.After removing repeated elements, your program should update array ***Arr,*** so that size of each row must be according to the total items presents in it.Sample run of RemoveRepetitionis shown below:

Before Function Call	**After Function Call**
Arr:	**Arr:**
4, 4, 4, 3, 3, 2,-1	4, 3, 2, -1
9, 9, 9, 6, 6, 5, 5, 5, 4, -1	9, 6, 5, 4, -1
6, 6, 6, 4, 4, 3, 3, -1	6, 4, 3, -1
1, 1, 1, -1	1, -1

Note. You are NOT ALLOWED to change the function header, you can consider that data of each row is sorted in descending order.

```
// Write code here
```