

NAME # S. Zulfigar Hasan : ID # 14937

### Sessional Assignment

Task # 1 Design Program (in any computer language of your choice) to implement the Algorithm to

⊗ Push an Element in a Queue.

```
int main()
```

```
{
```

```
stack <int> stack1; // empty stack of integer type
```

```
stack1.push(100);
```

```
stack1.push(200);
```

```
stack1.push(300);
```

```
stack1.push(400);
```

```
stack1.push(500);
```

```
cout << "THE topmost element of the stack is; " << stack1.top() << endl;
```

```
cout << "THE size of the stack is " << stack1.size() << endl;
```

```
if (stack1.empty())
```

```
{
```

```
cout << "stack is empty" << endl;
```

```
}
```

```
else
```

```
{
```

```
cout << "stack is not empty" << endl;
```

```
}
```

```
}
```

⊗ Pop an element from a Queue

begin procedure pop : stack

if stack is empty  
return null  
end if

data ← stack[top]

top ← top - 1

return data

end procedure

int pop (int data) {

if (! is empty ()) {

data = stack[top];

top = top - 1;

return data;

} else {

print f ("could not retrieve

data, stack is empty.\n");

};

}

Task # 2

Design a program (in any computer language of your choice) to implement the Algorithm to

# Create a one way linked list

```
# include < iostream >
```

```
# include < cstdio >
```

```
# include < cstdlib >
```

```
using namespace std;
```

```
/*
```

```
 * Node Declaration
```

```
 */
```

```
struct node
```

```
{
```

```
int info;
```

```
struct node * next;
```

```
} * start;
```

```
/*
```

```
 * #
```

```
 * class Declaration
```

```
 */
```

```
class single - list
```

```
{
```

```
public:
```

```
node * create_node (int);
```

```
void insert - begin ();
```

```
void insert - pos ();
```

```
void insert pos last ();
```

```
void delete - pos ();
```

```
void sort ();
```

```
void search ();
void update ();
void reverse ();
void display ();
Single = list ();
}
```

```
start = null;
```

```
};
```

\* Insert an element in a one way linked list

```
public class linked list main {
public static void main (string args[]) {
singly linked list my linked list =
new singly linked list ();
my linked list . insert First (5);
my linked list . insert First (6);
my linked list . insert First (7);
my linked list . insert First (1);
my linked list . insert Last (2);
Node node = new Node ();
node . data = 1;
my linked list . delete After (node);
my linked list . print linked list ();
}
```

```
} out put printing linked list
```

```
{1}
{6}
{7}
{1}
{2}
```