Slip 1

A) Write a C++ program to check maximum and minimum of two integer numbers. (Use Inline function and Conditional operator)

```
#include<iostream>
#include <conio.h>

inline int max(int a, int b) {
  return ((a > b) ? a : b);
}

inline int min(int a, int b) {
   return ((a < b) ? a : b);
}

int main() {
   int a, b;

   cout << "Enter 2 numbers" << endl;
   cout << "Number 1: ";
   cin >> a;
   cout << "Number 2: ";
   cin >> b;

   cout << "The maximum number is: " << max(a, b) << endl;
   cout << "The minimum number is: " << min(a, b) << endl;

   return 0;
}
```

Slip 2

A) Write a C++ program to find volume of cylinder, cone and sphere. (Use function overloading).

```
#include<iostream.h>
#include<conio.h>

float volume(int r, int h)  //Volume of Cylinder
{
   return (3.14*r*r*h);
}

float volume (float r, float h) //Volume of Cone
{
   return (3.14*r*r*h/3);
}

float volume(float r)   //Volume of Sphere
```

```
 {
   return (4/3*3.14*r*r*r);
}


int main()
{
   float co_h, co_r, sp_r;
   int cy_h, cy_r;

   cout << "Enter dimensions" << endl;

   cout << "1. Cylinder" << endl;
   cout << "Height: ";
   cin >> cy_h;
   cout << "Radius: ";
   cin >> cy_r;

   cout << endl;

   cout << "2. Cone" << endl;
   cout << "Height: ";
   cin >> co_h;
   cout << "Radius: ";
   cin >> co_r;

   cout << endl;

   cout << "3. Sphere" << endl;
   cout << "Radius: ";
   cin >> sp_r;

   cout << endl;

   cout << "The volume of Cylinder is: " << volume(cy_h, cy_r) << endl;
   cout << "The volume of Cone is: " << coneVol(co_h, co_r) << endl;
   cout << "The volume of Sphere is: " << volume(sp_r) << endl;

   return 0;
}
```

Slip 3

A) Write a C++ program to interchange values of two integer numbers. (Use call by reference).

```
#include<iostream.h>
#include<conio.h>

void swap(int &a, int&b)
{
   int temp;

   temp = a;
   a = b;
   b = temp;
}


int main()
{
   int a,b;


   cout << "Enter first number: "<< endl;
   cin >> a;
   cout << "Enter second number: "<< endl;
   cin >> b;

   cout <<"Numbers Before Swapping:" <<  "a = " << a << " b = " << b << endl;

   swap(a, b);

   cout << " Numbers After Swapping:" << "a = " << a << "b = " << b << endl;

   return 0;
}
```

Slip 4

A) Write a c++ program to accpet worker information Worker_Name, No_Of_Hours_Worked, Pay_Rate and Salary. Write necessary functions to calculate and display the salary of Worker. (Use Default values for Pay_Rate)

```
#include<iostream.h>
#include<conio.h>
```

```
class WorkerInformation
 {
    char Worker_Name[50];
    int No_Of_Hours_Worked, Pay_Rate, Salary;

    public:
        void acccept()
       {
            cout << "Enter name of the worker: ";
            cin >> Worker_Name;

            cout << "Enter number of hours worked: ";
            cin >> No_Of_Hours_Worked;

            cout << "Enter pay rate: ";
            cin >> Pay_Rate;
        }

        void display()
      {
            cout << endl << "Worker details" << endl;
            cout << "Name: " << Worker_Name << endl;
            cout << "Salary: " << calSal(No_Of_Hours_Worked, Pay_Rate) << endl;
        }

        int calSal(int work_hrs, int pay_rate=500)
       {
            return (work_hrs*pay_rate);
        }
};
```

```
int main()
{
   WorkerInformation w;
   w.acccept();
   w.display();
   return 0;
}
```

Slip 5

A) Consider the following C++ class

```
class Point {
   int x, y;

   public:
      void setpoint(int, int); //To set the values of x and y co-ordinate.
      void showpoint(); //To display co-ordinate of a point P in format
(x, y)
}
```

```
#include<iostream.h>
#include<conio.h>


class Point
{
   int x, y;


   public:
      void setpoint(int a, int b)
     {
         x = a;
         y = b;
      }
```

```
    void showpoint() {
      cout << "(" << x << ", " << y << ")";
    }
};


int main()
{
  int a, b;
  Point p;

  cout << "Enter coordinates" << endl;
  cout << "Enter x: ";
  cin >> a;
  cout << "Enter y: ";
  cin >> b;

  p.setpoint(a, b);
  p.showpoint();

  return 0;
}
```

Slip 6

A) Write a C++ program to create two Classes Square and Rectangle. Compare area of both the shapes using friend function. Accept appropriate data members for both the classes.

```
#include<iostream.h>
#include<conio.h>

class Square
{
   public:
      int s;

      void getdata()
     {
         cout << "Enter the side of the square: ";
         cin >> s;
      }

      int calArea()
     {
         return (s*s);
      }

      friend void compare(int s, int r);
};

class Rectangle
{
   public:
      int l, w;

      void getdata()
     {
         cout << "Enter the length of the rectangle: ";
         cin >> l;
         cout << "Enter the width of the rectangle: ";
```

```
            cin >> w;
        }

        int calArea()
       {
            return (l*w);
        }

        friend void compare(int s, int r);
};

void compare(int s, int r)
{
    if(s > r)
   {
        cout << "The area of square is bigger than area of rectangle." << endl;
    }
 else
{
        cout << "The area of rectangle is bigger than area of square." << endl;
    }
}



int main()
{
    int s_area, r_area;
    Square s1;
    Rectangle r1;

    s1.getdata();
    s_area = s1.calArea();


    r1.getdata();
    r_area = r1.calArea();


    cout << "Square: " << s_area << endl;
    cout << "Rectangle: " << r_area << endl;


    compare(s_area, r_area);


    return 0;
}
```

Slip 7

```
#include<iostream.h>
#include<conio.h>

class slip7a
{
 private:
       char str[50];
public:
       int replace(char ch1, char ch2='b')
      {
           int i,cnt=0;
          cout<<"enter string:";
           cin>>str;
           for(i=0;str[i]!='\0';i++)
          {
              if(str[i]==ch1)
             {
                 str[i]=ch2;
                 cnt++;
             }
          }
      cout<<"After replacement string is"<<str<<endl;
      return cnt;
```

```
   }
};

int main()
{
slip7a ob1;
int num;
num=ob1.replace();
cout<<"Number of replacements are:"<<num<<endl;
return 0;
}
```

Slip 8

A) Write a C++ program to create a class Number, which contain static data member 'cat' and member function 'Display()'. Display() should print number of times display operation is performed irrespective of the object responsible for calling Display().

```
#include<iostream.h>
#include<conio.h>

class Number
{
static int cnt;
int n;
public:
void Display()
{
 cnt++;
cout<<"Number of times display operation performed is:"<<cnt<<endl;
}
};
int Number::cnt=0;
int main()
{
Number ob1,ob2,ob3;
ob1.Display();
ob2.Display();
ob3.Display();
ob1.Display();
return 0;
}
```

Slip 9

A) Consider the following C++ class

```
class Person
{
char Name [20];
char Add r [30];
float Salary:
float tax amount;
public:
//member functions
};
```

Calculate tax amount by checking salary of a person

For salary-20000 tax rate-0

•For salary>20000 ||<-40000 tax rate-5% of salary.

For salary>40000 tax rate 10% of salary.

```
#include<iostream.h>
#include<conio.h>

class person
{
 char addr[20];
 float sal,tax;
public:
   void get()
```

```
{
cout<<"Enter the name, address, salary: \n";
cin>>name>>addr>>sal:
}
void put()
{
cout<<"Person Information:\n":
cout<<"Name tAddress Salary Tax: \n";

cout<<name<<"\t"<<addr<<"\t"<<sal<<"\t"<<tax<<endl;
}
void cal tax()
{
If(sal<=20000)
{
  tax=0;
}
else if( sal>=20000/sal<=40000)
{
    Tax=(sal*5)/100;
}
else if(sal>40000)
{
   Tax=(sal*10)/100;
}
}
};

Void main()
{
  person p;
```

```
clrscr();

p.get();

p.cal_tax();

p.put();

getch();

}
```

Slip 10

A) Write a C++ program to create a class Account with data members Acc_number, Acc_type and Balance. Write member functions to accept and display 'n' account details. (Use dynamic memoryallocation).

```
#include<iostream.h>
#include<conio.h>
#include<stdlib.h>

 class Account
{
  private:
           int Acc_no,Balance;
           char Acc_type[30];
   public:
     void get_data()
    {
        cout<<"\n Enter Acc_no.:";
       cin>>Acc_no;

        cout<<"\n Enter Acc_type :";
        cin>>Acc_type;

         cout<<"\n Enter Balance :";
          cin>>Balance;
    }

void display_data()
{
cout<<"\t"<<Acc_no<<"\t"<<"\t"<<Acc_type<<"\t"<<Balance;
}
};

int main()
{
 clrscr();
int num;

 Account* a = new Account[4];

 cout<<"\n How many records u want?: ";
cin>>num;

 for(int i=0;i<num;i++)
{
a[i].get_data();
}

for(i=0;i<num;i++)
{
a[i].display_data();
}

return 0;

 }
```