PART II VISCOOSI - KRUNAL RANK 1/8

Ans 1:

```
template<class T>
class vector{
    T* v;
    int size;
    public:
        void create(int);
        void modify(int, T);
        void mult(int);
        void display();
};
template<class T>
void vector<T>::create(int s){
    v = (T*) malloc(s*sizeof(T));
    size = s;
}
template<class T>
void vector<T>::modify(int i, T val){
    if(i>=size) return;
    v[i] = val;
}
void vector<T>::mult(int s){
    for(auto &i: v) i*=s;
}
void vector<T>::display(){
std::cout<<"(";
    for(int i=0; i<size; i++){
        if(i=size-1){ std::cout<<v[i]; break;}
  std::cout<<v[i]<<",";
    }
    std::cout<<")"<<std::endl;
}
```

Ans 2:

a) Yes class D requires constructor even if it doesn't contain any data member of its own because this constructor will be used to initialise data members of its parent class B if they exist in the parent class.

b) The protected keyword specifies access to data members in the member list upto next access specifier (private or public).
Data members declared as protected can only be accessed within the package (or class) or outside class only via inheritance. They cannot be accessed in any other way.

Ans 3:

```
class test {
    private :
        int m;
    public :
        void getdata () {
            cout << " Enter Number: ";
            cin >> m;
        }
        void display () {
            cout << m;
        };
};      // missing parenthesis and semicolon
int main () {       // missing int return type of main method
    test * T ;      // T should be of type Test pointer
    T -> getdata();
```

```
T→ display();
test *p;
p→ getdata();      //Pointer type uses arrow notation
                   // to access class methods

(*p).display();
}
```

Ans 3. i)

```
class SB;
class SM {
    double m,cm;
    public:
      void SM(double, double);
      void display();
      friend SM add(SM, SB);
};
class SB {
    int feet;
    double inches;
    public:
        SB(int, double);
      void display();
       friend SB add(SM, SB);
}
SM::SM(double M, double CM) { m=M; cm=CM; }
SM::display() {
    std::cout<<m<<"metres"<<cm<<"centimeters"<<std::endl;
}
SM SM::add(SM, SB b) {
    m = m + cm/100.0 + b.inches*0.0254 + b.feet*0.304;
    cm = m*100;
```

```
    return a;
}

SB :: SB (int F, double I){
    feet = F;
    inches = I;
}
SB :: display(){
  std:: cout<< "Feet : " << feet <<std:: endl;
  std:: cout << "Inches :" <<inches<<std::endl;
}
SB :: add(SM a){
   feet = feet + a.m * 3.2808;
   inches = a.m
   a.m = a.m + a.cm /100 + inches * 0.025 + feet * 0.3048;
   a.cm = a.m * 100;
   feet = a.m * 3.2808;
   inches = a.m * 39.37;
}
```

Sample Output :-

4.6002 m 460.02 cm

Feet : 5

Inches : 3

PART II:- U18CO051-KRUNAL RANK 5/8

ii)
```
#include <bits/stdc++.h>
using namespace std;
class books {
    public:
    string author, title, publisher; vector<Book*> books_available;
    double price;
    int copies, stock;
    void setData() { string author, title, publisher; double price; int copies, stock;
        cout<< "Enter following details: " <<endl;
        Cout<<" Author:- ";
        cin>> author;
        cout<<" Title :- ";
        cin>>title
        cout<<" Publisher :- ";
        Cin>> publisher;
        cout<<" Price :- ";
        cin>> price;
        cout <<" Copies :- ";
        Cin >> copies;
        stock =1;
        b.push_back(new Book(Author, Title, Publisher, Price, Copies, Stock);
    }
    void check() {
        string t, a;
        cout<< "Search for books: " << endl;
        cout<< " Enter Title :" <<endl;
        cout<<" En cin>>t;
        cout << " Enter Author: " <<endl;
        cin>> a;
        Book* b;
        for (Book* it : books_available) {
```

```
if (it->title == t && it->author == a){
    b = it;
    break;
}
}
if(!b){
    cout<<"Sorry! Book not available"<<endl;
    return;
}
cout<<"Book available :- "<<endl;
cout<<"Author :- "<<b->author<<endl;
cout<<"Title :- "<<b->title<<endl;
cout<<"Publisher :- "<<b->publisher<<endl;
cout<<"Price :- "<<b->price<<endl;
cout<<"Copies :- "<<b->copies<<endl;
cout<<"Stock :- "<<b->stock<<endl;
cout<<"Enter Required copies :- ";
int c;
cin>>c;
if(c<=b->copies){
    cout<<"Total price :- "<<b->price*c<<endl;
}
return;
}
class Book{ string author,publisher,title; int stock,copies; double price;
public:
    Book(string A,string T, string P, double pri, int C,int S){
        author=A; title=T; publisher=P; price=Pri; copies=C;
        stock=S;
    }
};
```

PART II U1820081 _KRUNAL RANK 7/8

Sample output:-

Enter following details:-
Enter Author :- Anvi
Enter Title :- Tushar
Enter Publisher :- Pushkar
Enter Price :- 20.00
Enter Copies :- 100

Search for Books:-
Enter Title :- Tushar
Enter Author :- Anvi
Book Available
Author :- Anvi
Title :- Tushar
Publisher :- Pushkar
Price :- 20.00
Copies :- 100
Stock :- 1
Enter Required Copies :- 20
Total Price :- 400

```
int main()
{
    books * bookstore;
    bookstore->setData();
    bookstore -> check();
}
```

kstore;
Data();
check();

iii) class Account