**/\* Program No. :**

**Aim : WAP to create two classes DM and DB which store the value of distances. DM stores distances in metres and centimetres and DB in feet and inches and both the distances and shows the result in both formats.**

**\*/**

#include<iostream.h>

#include<conio.h>

class dm

{

float metre,centimetre;

public:

dm(float);

dm(int);

dm(int,int);

int get\_centimetre();

float get\_metre();

void display() const;

};

class db

{

float feet,inches;

public:

db(float);

db(int);

db(int,int);

int get\_inches();

float get\_feet();

void display() const;

};

dm::dm(int a)

{

metre=a/100;

centimetre=a%100;

}

dm::dm(float a)

{

metre=(int)a;

centimetre=(a-metre)\*100;

}

dm::dm(int a,int b)

{

metre=a;

centimetre=b;

}

int dm::get\_centimetre()

{

return((metre\*100)+centimetre);

}

float dm::get\_metre()

{

return(metre+(centimetre/100));

}

void dm::display() const

{

cout<<(int)metre<<" m and "<<(int)centimetre<<" cm";

}

db::db(int a)

{

feet=a/12;

inches=a%12;

}

db::db(float a)

{

feet=(int)a;

inches=(a-feet)\*12;

}

db::db(int a,int b)

{

feet=a;

inches=b;

}

int db::get\_inches()

{

return((feet\*12)+inches);

}

float db::get\_feet()

{

return(feet+(inches/12));

}

void db::display() const

{

cout<<(int)feet<<" feet and "<<(int)inches<<" inches";

}

void main()

{

clrscr();

dm m1(548);

db b1(74);

dm m2((int)(m1.get\_centimetre()+(b1.get\_inches()\*(30.48/12))));

db b2((float)(m1.get\_metre()\*(3.2808))+(b1.get\_feet()));

cout<<"\n\n\t\tDistance in DM : ";

m1.display();

cout<<"\n\n\t\tDistance in DB : ";

b1.display();

cout<<"\n\n\n\t\tTotal Distance in DM : ";

m2.display();

cout<<"\n\n\t\tTotal Distance in DB : ";

b2.display();

getch();

}

**/\***

**Name : Rohit Aggarwal**

**Roll No. : 7CS-097**

**\*/**