

C++ Programming Assignment - 30

1. Why we can not create object of such a class which contains pure virtual function in it?
2. What is mean by pure virtual function?
3. What happens if base class contains virtual function under private access specifier?
4. What is mean by Abstract method and Concrete Method?
5. Explain below syntax and draw its diagrammatic representation.

```
class Base
```

```
{
```

```
  public :
```

```
    int i;
```

```
    float f;
```

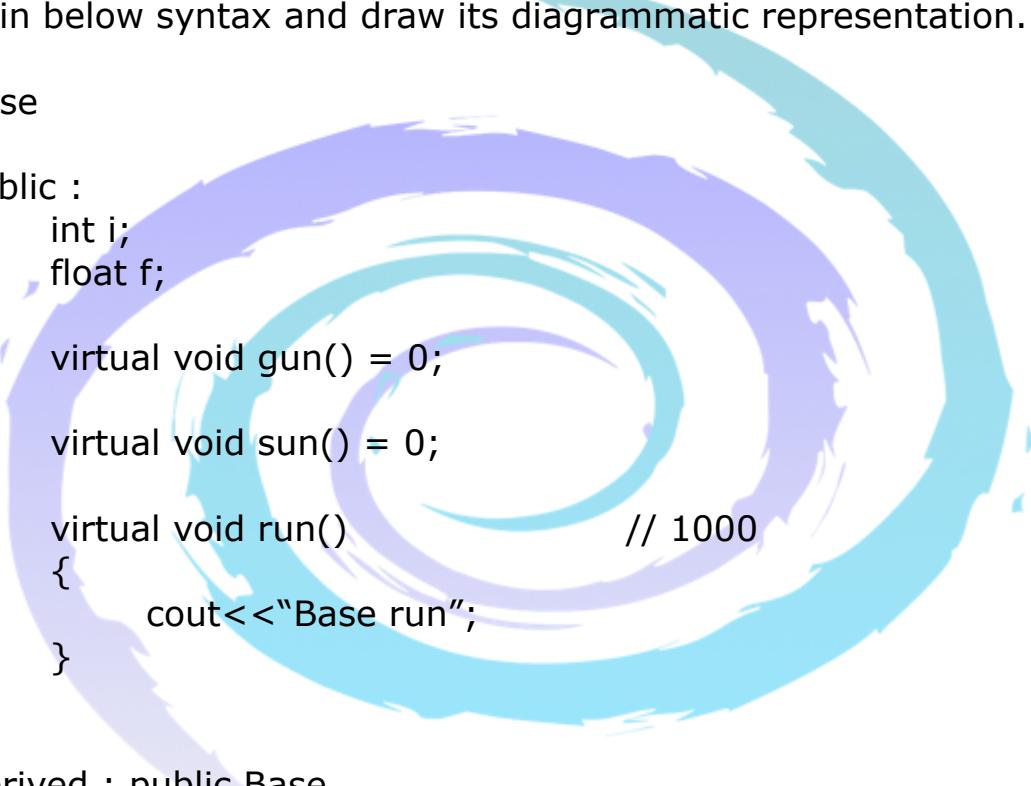
```
    virtual void gun() = 0;
```

```
    virtual void sun() = 0;
```

```
    virtual void run()
```

```
{
```

```
      cout<<"Base run";
```



```
// 1000
```

```
}
```

```
};
```

```
class Derived : public Base
```

```
{
```

```
  public :
```

```
    int i;
```

```
    double d;
```

```
    void sun() // 2000
```

```
{
```

```
      cout<<"Derived sun";
```

```
}
```

```
    void fun() // 3000
```

```
{
```

```
      cout<<"Derived fun";
```

```
}
```

```

void gun()           // 4000
{
    cout<<"Derived gun";
}

virtual void mun()   // 5000
{
    cout<<"Derived mun";
}

};

int main()
{
    Base *bp = NULL;
    Derived dobj;

    bp = &dobj;

    bp -> fun();
    bp -> gun();
    bp -> sun();
    bp -> run();
    bp -> mun();

    return 0;
}

```

6. Draw the memory layout and draw the assembly instructions of function calls.

```

class Base
{
public :
    int i;
    float f;

    void fun()           // 1000
    {
        cout<<"Base fun";
    }

    virtual void gun()   // 2000
    {
        cout<<"Base gun";
    }
};

```

```

class derived : public base
{
public :
    int i;
    double d;

    virtual void fun()          // 3000
    {
        cout<<"Derived fun";
    }

    void gun()                  // 4000
    {
        cout<<"Derived gun";
    }

    virtual void sun()          // 5000
    {
        cout<<"Derived sun";
    }
};

int main()
{
    Base *bp = new Derived;
    bp->gun();

    return 0;
}

```

7. Draw the objects layout of below syntax.

```

class base1
{
public :
    int i;
    float f;

    virtual void gun() = 0;
    virtual void sun() = 0;
    virtual void run()      // 1000
    {}

};

```

```

class base2
{
public :
    int j;
    float g;

    virtual void mun() = 0;

    virtual void fun() = 0;

    void fun() // 2000
    {}

};

class derived : public base1, base2
{
public :
    int i;
    double d;

    void sun() // 3000
    {}

    void fun() // 4000
    {}

    void gun() // 5000
    {}

    void mun() // 6000
    {}

};

int main()
{
    derived dobj;

    return 0;
}

```

8. What are the ways in which we can achieve upcasting in object oriented language?
9. What is the use of pure virtual function?
10. Can we create pointer of a class which contains pure virtual function in it?