Saransh Agarwal
RA1511003010320
Computer Science and Engineering

AIM : TO implement Classes and object and create a program in C++

Concepts used

1. **Inheritance**
2. **Constructor**
3. **Virtual function**
4. **Friend function**
5. **Inline function**
6. **Deconstructor**
7. **Virtual inheritance**
8. **Overloading**
9. **File stream**
10. **Polymorphism**

Program :

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

using namespace std;
int xx;
static long long price=0;
class smart_phone;
class laptop;
class product;
class gadget
{
        private :
                float screen_size;//inches
                string processor;
        public:
                char brand;
                gadget()
                {
                        cout<<"$Welcome to Custom Gadget shop$\n"<<endl;
                }
                friend int display(product p);
                void get_screen_size()
                {
```

```
		cout<<"Screen Size : ";
		ifstream outfile;
		outfile.open("FILE.txt",ios::in);
		outfile>>screen_size;
		cout<<screen_size<<" Inches"<<endl;
		outfile.close();
	}
	void get_processor()
	{
		cout<<"Processor : ";
		ifstream outfile;
		outfile.open("FILE.txt",ios::in);
		outfile>>processor;
		outfile>>processor;
		cout<<processor<<endl;
		outfile.close();
	}
	void set_screen_size(float x)
	{
		screen_size=x;
		price = price + x*1000;
	}
	void set_processor()
	{
		int x;
		cout<<"Which Processor ?\n";
		cout<<"1. Snapdragon\n2. Xynos\n3. Mediatech\n4. Intel\n5. AMD\n";
		cin>>x;
		switch(x)
		{
			case 1 :
				processor = "Snapdragon";
				price=price + 10000;
				break;
			case 2 :
				processor = "Xynos";
				price=price + 8000;
				break;
			case 3 :
				processor = "Mediatech";
				price=price + 5000;
				break;
			case 4 :
				processor = "Intel";
				price=price + 30000;
				break;
			case 5 :
```

```
                                        processor = "AMD";
                                        price=price + 20000;
                                        break;
                                default :
                                        processor = "Intel";
                                        break;
                        }
                        fflush(stdin);
                        ofstream outfile;
                        outfile.open("FILE.txt",ios::app);
                        outfile<<processor<<endl;
                        outfile.close();
                }
};
class smart_phone : public virtual gadget
{
        private :
                int ram;//GB
                int storage;//GB
                int camera;//Mega Pixel
        public :
                virtual int choice();
                virtual int print()
                {
                        ifstream outfile;
                        outfile.open("FILE.txt",ios::in);
                        outfile>>ram;
                        cout<<"RAM : "<<ram<<" GB"<<endl;
                        outfile>>storage;
                        cout<<"Storage : "<<storage<<" GB"<<endl;
                        outfile>>camera;
                        cout<<"Camera : "<<camera<<" MP"<<endl;
                        outfile.close();
                }
};
class laptop : public virtual gadget
{
        private :
                int ram;//GB
                int storage;//TB
                int graphics;//GB
        public :
                virtual int choice();
                virtual int print()
                {

                        ifstream outfile;
```

```
                outfile.open("FILE.txt",ios::in);
                outfile>>ram;
                cout<<"RAM : "<<ram<<" GB"<<endl;
                outfile>>storage;
                cout<<"Storage : "<<storage<<" GB"<<endl;
                outfile>>graphics;
                cout<<"Graphics : "<<graphics<<" GB"<<endl;
                outfile.close();
        }
};
class product : public smart_phone, public laptop
{
    private :
        int model_id;
    public :
        inline int discount();
        int get_id()
        {
            cout<<"\n\n!!$ Your Custom Product is ready $!!\n";
            cout<<"Model ID : ";
            cout<<"GS"<<price/1000+6<<"FX"<<price%1000<<endl;
        }
};
int discount()
{
    int profit;
    profit = price/13;
    cout<<"\nYou recieved a discount of Rupees = "<<profit<<endl;
    cout<<"Final Price to be paid : "<<price-profit<<endl;
}
int display(product p)
{
    p.get_id();
    p.get_screen_size();
    p.get_processor();
    if(xx==1)
    {
        p.smart_phone::print();
    }
    else if(xx=2)
    {
        p.laptop::print();
    }
    cout<<"Total Price is Rupees : "<<price;
    discount();
}
int smart_phone :: choice()
```

```
{
        int x;
        cout<<"How much RAM ?\n";
        cout<<"1. 2 GB\n2. 3 GB\n3. 4 GB\n";
        cin>>x;
        switch(x)
        {
                case 1 :
                        ram=2;
                        break;
                case 2 :
                        ram=3;
                        break;
                case 3 :
                        ram=4;
                        break;
                default :
                        ram = 8;
                        break;
        }
        fflush(stdin);
        ofstream outfile;
        outfile.open("FILE.txt",ios::app);
        outfile<<ram<<endl;
        outfile.close();
        price=price + ram*1000;
        cout<<"How much Storage in GB?\n";
        cout<<"1. 32 GB\n2. 64 MGB\n3. 128 GB\n";
        cin>>x;
        switch(x)
        {
                case 1 :
                        storage = 32;
                        break;
                case 2 :
                        storage = 64;
                        break;
                case 3 :
                        storage = 128;
                        break;
                default :
                        storage = 1000;
                        break;
        }
        fflush(stdin);
        outfile.open("FILE.txt",ios::app);
        outfile<<storage<<endl;
```

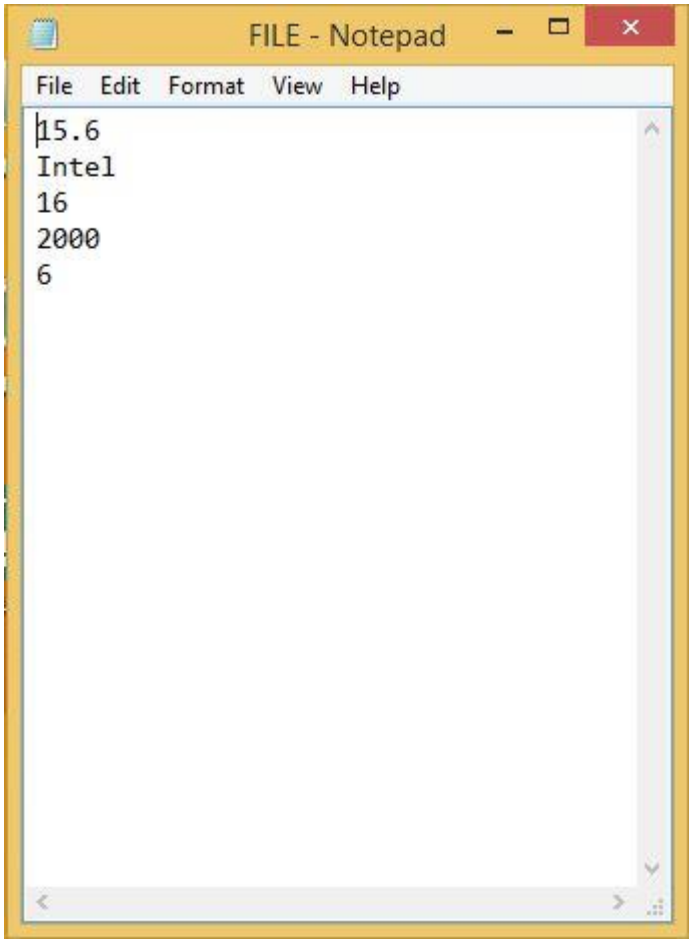
```
                outfile.close();
                price=price + storage*5;
                cout<<"Camera mega pixel ?\n";
                cout<<"1. 8 MP\n2. 16 MP\n3. 23 MP\n";
                cin>>x;
                switch(x)
                {
                        case 1 :
                                camera=8;
                                break;
                        case 2 :
                                camera=16;
                                break;
                        case 3 :
                                camera=23;
                                break;
                        default :
                                camera=16;
                                break;
                }
                fflush(stdin);
                outfile.open("FILE.txt",ios::app);
                outfile<<camera<<endl;
                outfile.close();
                price=price + camera*1000;
        }
int laptop:: choice()
        {
                int x;
                cout<<"How much RAM ?\n";
                cout<<"1. 4 GB\n2. 8 GB\n3. 16 GB\n";
                cin>>x;
                switch(x)
                {
                        case 1 :
                                ram=4;
                                break;
                        case 2 :
                                ram=8;
                                break;
                        case 3 :
                                ram=16;
                                break;
                        default :
                                ram = 8;
                                break;
                }
```

```
fflush(stdin);
ofstream outfile;
outfile.open("FILE.txt",ios::app);
outfile<<ram<<endl;
outfile.close();
price=price + ram*2000;
cout<<"How much Storage in GB?\n";
cout<<"1. 500 GB\n2. 1000 MGB\n3. 2000 GB\n";
cin>>x;
switch(x)
{
        case 1 :
                storage = 500;
                break;
        case 2 :
                storage = 1000;
                break;
        case 3 :
                storage = 2000;
                break;
        default :
                storage = 1000;
                break;
}
fflush(stdin);
outfile.open("FILE.txt",ios::app);
outfile<<storage<<endl;
outfile.close();
price=price + storage*10;
cout<<"Graphic Card Size  ?\n";
cout<<"1. 2 GB\n2. 4 GB\n3. 6 GB\n";
cin>>x;
switch(x)
{
        case 1 :
                graphics=2;
                break;
        case 2 :
                graphics=4;
                break;
        case 3 :
                graphics=6;
                break;
        default :
                graphics=1;
                break;
}
```

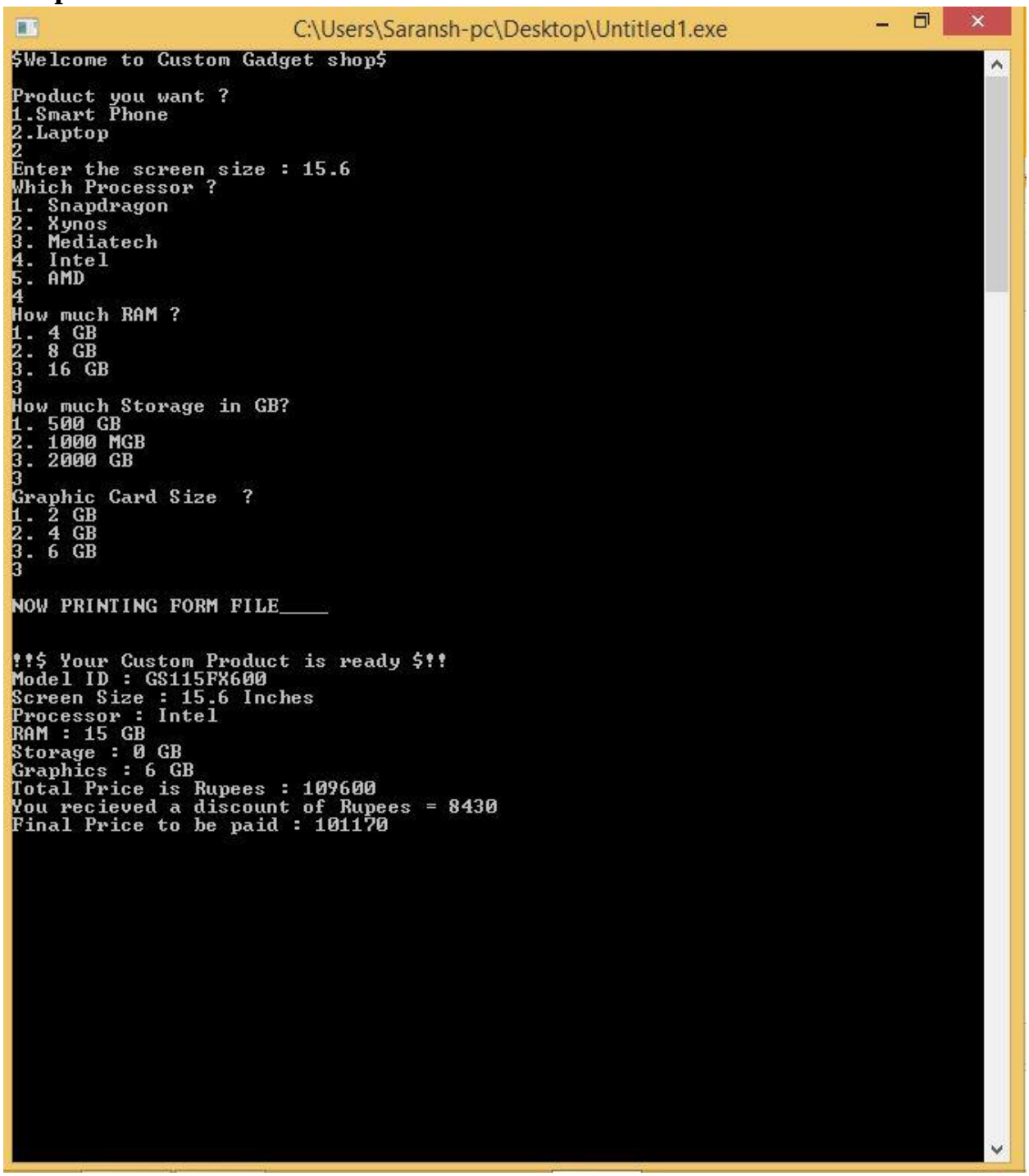
```
                        fflush(stdin);
                        outfile.open("FILE.txt",ios::app);
                        outfile<<graphics<<endl;
                        outfile.close();
                        price=price + graphics*2000;
                }
int main()
{
        ofstream outfile;
        float screen;//screen_size
        register int x;
        product p;
        cout<<"Product you want ?\n1.Smart Phone\n2.Laptop\n";
        cin>>x;
        switch(x)
        {
                case 1:
                        xx=1;
                        cout<<"Enter the screen size : ";
                        cin>>screen;
                        outfile.open("FILE.txt",ios::out);
                        outfile<<screen<<endl;
                        outfile.close();
                        p.set_screen_size(screen);
                        p.set_processor();
                        p.smart_phone::choice();
                        break;
                case 2:
                        xx=2;
                        cout<<"Enter the screen size : ";
                        cin>>screen;
                        outfile.open("FILE.txt",ios::out);
                        outfile<<screen<<endl;
                        outfile.close();
                        p.set_screen_size(screen);
                        p.set_processor();
                        p.laptop::choice();
                        break;
        }
        cout<<"\nNOW PRINTING FORM FILE____\n";
        display(p);

        getch();
}
```

File Created During Implementation :



Output :



Result : The program successfully compiled and executed. All the concepts were applied and worked correctly