NATIONAL PUBLIC SCHOOL
COMPUTER SCIENCE PROJECT
ARMS REGISTRY

Name: Sriram R

Class: XII

Register Number:

ACKNOWLEDGEMENT

I would sincerely like to thank National Public School, Rajajinagar for giving me an opportunity to work on this project. I would also like to thank Mrs. Bhargavi for her help and guidance. Her constant support and encouragement has enabled me to complete this project successfully.

I would also like to extend my sincere thanks to Mrs. Poornima and Mrs. Tanushree who have been very helpful and have assisted me with the project.

INDEX

TOPIC	PAGE NO.
About C++	5
About Project	6
Header Files	7
Files Used	8
Classes Used	9
Flow of Control	11
Source Code	12
Output	56
References	72

ABOUT C++

C++ is an Object Oriented Programming (OOP) language, developed by Bjarne Stroustrup and is an extension of C language. It is therefore possible to code C++ in a "C Style" or "Object-Oriented style". In certain scenarios, it can be coded in either way and is thus an effective example of a hybrid language.

C++ is a general purpose object oriented programming language. It is considered to be an intermediate level language, as it encapsulates both high level and low level language features. Initially, the language was called, "C with Class" as it had all the properties of C language, but with an additional concept of 'classes'. However, it was renamed to 'C++' in 1983.

ABOUT THE PROJECT

This project entitled 'Arms Registry' makes use of concepts such as 'File Handling' and 'Object Oriented Programming' in C++.

These concepts enable features such as addition and deletion of weapon details in the catalogue and printing of final receipt among other functions.

The project aims to create a better understanding of online weapons purchasing. It also gives me a chance to explore firearms and gave me an excellent opportunity to understand the importance of a friendly and clutter-free User Interface.

HEADER FILES

#include <iostream>

#include <fstream>

#include <process.h>

#include <string.h>

#include <stdlib.h>

#include <stdio.h>

#include <ctype.h>

#include <conio.h>

#include <dos.h>

FILES USED

PRODUCT.dat – Binary File

Temp.dat – Binary File

BILL.dat – Binary File (For final receipt)

CLASSES USED:

(with member functions)

1. <u>Menu</u>(class)

  ● main_menu()   -Main Menu that contains all functions
  ● edit_menu()   -provides option to add and delete items to list

2. <u>Product</u>(class)

  ● add();   -adds new items to the catalogue
  ● delet();   -deletes selective item from catalogue
  ● list();   -displays entire catalogue
  ● buy();   -allows purchase of item based on code

Public members:

  ● delete_record(int);   -deletes item from catalogue
  ● insert(int);
  ● display_record(int);   -displays cumulative of all purchase
    sessions
  ● found(int); - int   -function to check if code input matches
                               pre-existing record
  ● recordno(int); - int   -function that keeps track of all the records
                               by assigning codes to each item
  ● sort();   -sorts records on the basis of code of item
  ● code; - int
  ● name[30]; - char
  ● cost; - float
  ● price; - float

3. <u>Account</u>(class)

- code - int
- billno - int
- length - int
- dd - int
- mm - int
- yy - int
- cost - float
- price - float
- quantity - float
- name[30]; - char

Public:

- bill_list();
- prepare_bill(int);
- last_billno();
- add_bill();

FLOW OF CONTROL
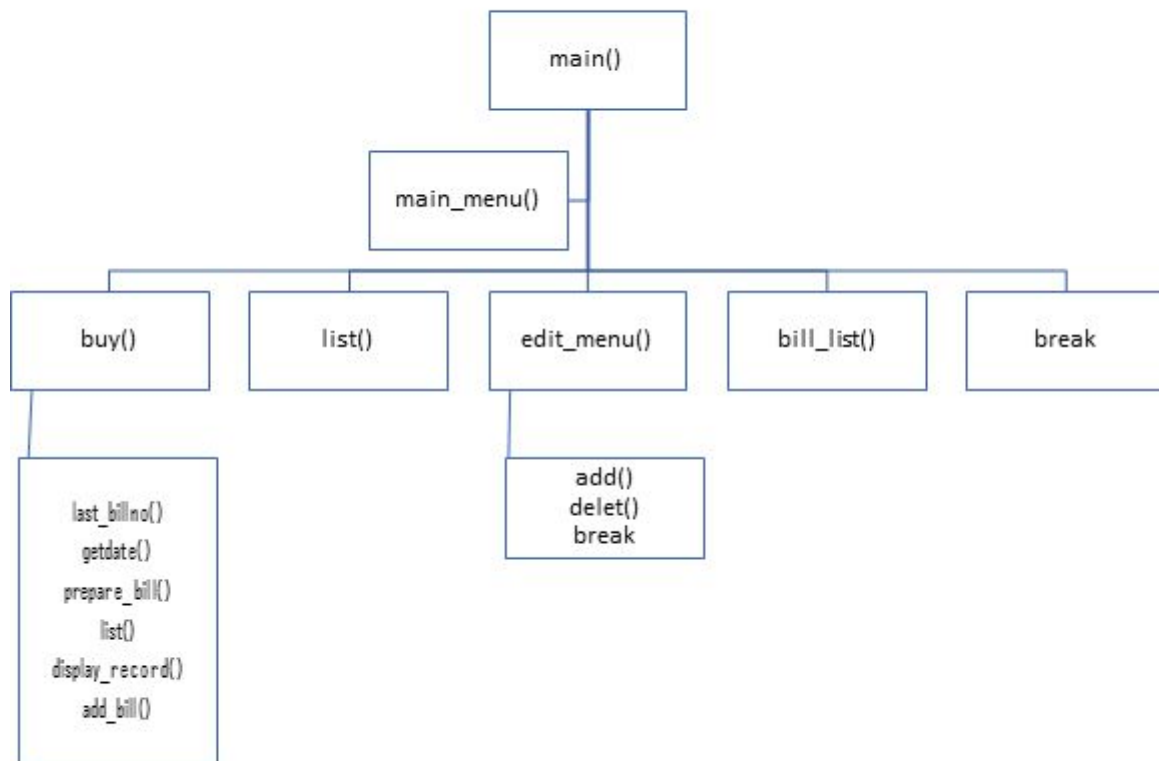
Source Code

```cpp
// PROJECT FINAL

#include <iostream.h>

#include <fstream.h>

#include <process.h>

#include <string.h>

#include <stdlib.h>

#include <stdio.h>

#include <ctype.h>

#include <conio.h>

#include <dos.h>


class menu
{
 public :

       void main_menu();


private :

       void edit_menu();
} ;



class product
{
```

```cpp
public :

        void add();

        void delet();

        void modify();

        void list();

        void buy();

private :

        int last_code();

        void delete_record(int);

        void insert(int);

        void display_record(int);

        int found(int);

        int recordno(int);

        void sort();


        int code;

        char name[30];

        float cost, price;

} ;
```

/* CLASS NAME : Account

 DETAILS : IT CONTROLS ALL THE FUNCTIONS

 RELATED TO MAKING BILL

*/

```cpp
class account
{
public :
        void bill_list();

        void prepare_bill(int);

        int last_billno();

        void add_bill(int, int t_itemcode, char *t_itemname, float t_qty,
float t_cost, float t_price);
private :
        int code, billno, length;

        int dd, mm, yy;

        float cost, price, quantity;

        char name[30];
} ;



void menu :: main_menu()
{
        clrscr();

        char ch;

        while (1)

        {
                clrscr();

                gotoxy(10,3);

                cout<<"---------------------------------------------------------";
```

```cpp
gotoxy(10,23);
cout<<"----------------------------------------------------------";
gotoxy(28,6);
cout<<" A M M U - N A T I O N ";
gotoxy(28,7);
cout<<"----------------------";
gotoxy(30,9);
cout<<"1: BUY Weapon";
gotoxy(30,11);
cout<<"2: CATALOGUE";
gotoxy(30,13);
cout<<"3: EDIT PRODUCTS FILE";
gotoxy(30,15);
cout<<"4: PRINT RECEIPT";
gotoxy(30,17);
cout<<"0: QUIT";
gotoxy(30,20);
cout<<"Enter Your Choice : ";
ch = getche();
if (ch == '1')
{
            product p;
            p.buy();
}
else if (ch == '2')
{
```

```cpp
                    product p;
                    p.list();
        }
        else if (ch == '3')
            edit_menu();
        else if (ch == '4')
        {
                    account a;
                    a.bill_list();
        }
        else if (ch == '0')
                break;
    }
}


void menu :: edit_menu()
{
    char ch ;
    while (1)
    {
        clrscr() ;
        gotoxy(10,3);
        cout<<"------------------------------------------------------------";
        gotoxy(10,23);
        cout<<"------------------------------------------------------------";
```

```
gotoxy(34,6);
cout<<"E D I T   M E N U ";
gotoxy(32,7);
cout<<"---------------------";
gotoxy(32,5);
cout<<"---------------------";
gotoxy(32,10);
cout<<"1. A D D ";
gotoxy(32,14);
cout<<"2. D E L E T E ";
gotoxy(32,18);
cout<<"0. E X I T ";
gotoxy(32,20);
cout<<"Enter Choice : ";
ch = getche();


if (ch == '1')
{
            product p;
            p.add();
            break;
}


else if (ch == '2')
{
            product p;
```

```cpp
                        p.delet();

                        break;

            }

            else if (ch == '0')

                        break;

        }

}




int product :: last_code()

{

        fstream file;

        file.open("PRODUCT.DAT", ios::in);

        file.seekg(0,ios::beg);

        int t=0;


        while (file.read((char *) this, sizeof(product)))

                t = code;


        file.close();

        return t;

}
```

```cpp
void product :: list()
{
        clrscr();
        fstream file;
        file.open("PRODUCT.DAT", ios::in);
        file.seekg(0);
        int row = 6 , found = 0;

        gotoxy(25,2);
        cout <<"LIST OF ITEMS";

        gotoxy(3,4);
        cout <<" CODE     NAME        COST        PRICE";

        gotoxy(4,5);
        cout <<"**********************************************";

        while (file.read((char *) this, sizeof(product)))
        {
                //delay(20);
                found = 1;
                gotoxy(5,row);
                cout<<code;
                gotoxy(14,row);
                cout<<name;
                gotoxy(30,row);
```

```cpp
                cout<<cost;
                gotoxy(45,row);
                cout<<price;
                if (row==22)
                {
                        row = 5;

                    gotoxy(1,25);
                        cout <<"Press any key to continue...";
                        getche();    //Registers any key-press
                        clrscr();

                    gotoxy(30,2);
                        cout <<"LIST OF ITEMS";

                    gotoxy(3,4);
                        cout <<"ITEM CODE    ITEM NAME    ITEM
COST   ITEM PRICE";

                    gotoxy(2,5);
                        cout <<"----------------------------------------------";
                }
                else
                        row++;
        }
        if (!found)
```

```
        {
                gotoxy(5,10);

                cout<<"\n Records not found";

        }

        gotoxy(1,25) ;

        cout <<"\n Press any key to continue." ;

        getche() ;

        file.close () ;

}


// THIS FUNCTION ADDS RECORDS IN PRODUCT.DAT


void product :: add()

{

        int tcode, valid ;

        char ch, t_cost[10], t_price[10] ;

        tcode = last_code() ;

        tcode++ ;

        do

        {

                clrscr() ;

                gotoxy(71,2) ;

                cout <<"<0>=Exit" ;

                gotoxy(32,3) ;

                cout <<"ADDITION " ;        //Indented. Use this as Model

                gotoxy(26,4) ;
```

```
cout <<"-------------------" ;
gotoxy(5,6) ;
cout <<"Code : " <<tcode ;
gotoxy(5,8) ;
cout <<"Name : " ;
gotoxy(5,10) ;
cout <<"Cost : " ;
gotoxy(5,12) ;
cout <<"Price : " ;
do
{
        valid = 1 ;
        gotoxy(1,8) ;
        clreol() ;     //???
        gotoxy(1,24) ;
        clreol() ;     //???
        gotoxy(1,25) ;
        clreol() ;     //???
        gotoxy(3,25) ;
        cout <<"ENTER Name TO BE ADDED" ;
        gotoxy(5,8) ;
        cout <<"Name : " ;
        gets(name) ;
        strupr(name) ;
        if (name[0] == '0')
                return ;
```

```
                    if ((strlen(name) < 1) || (strlen(name) > 20))

                    {

                            valid = 0 ;

                            gotoxy(3,24) ;

                            cout <<"\n Input a shorter name ..." ;

                            getch() ;

                    }

        }

        while (!valid) ;

        do

        {

                    valid = 1 ;

                    gotoxy(1,10) ;

                    clreol() ;

                    gotoxy(1,24) ;

                    clreol() ;

                    gotoxy(1,25) ;

                    clreol() ;

                    gotoxy(3,25) ;

                    cout <<"ENTER COST TO ADD IN THE MENU" ;

                    gotoxy(5,10) ;

                    cout <<"Cost : " ;

                    gets(t_cost) ;

                    cost = atof(t_cost) ;

                    if (t_cost[0] == '0')

                            return ;
```

```
}
while (!valid) ;
do
{
        valid = 1 ;
        gotoxy(1,12) ;
        clreol() ;
        gotoxy(1,24) ;
        clreol() ;
        gotoxy(1,25) ;
        clreol() ;
        gotoxy(3,25) ;
        cout <<"ENTER PRICE TO ADD IN THE MENU"
;
        gotoxy(5,12) ;
        cout <<"Price : " ;
        gets(t_price) ;
        price = atof(t_price) ;


    if (t_price[0] == '0')
                return ;
}
while (!valid) ;
do
{
        gotoxy(1,15) ;
```

```
                clreol() ;

                gotoxy(1,24) ;

                clreol() ;

                gotoxy(1,25) ;

                clreol() ;

                gotoxy(5,15) ;

                cout <<"Do you want to save this record (y/n) : " ;

                ch = getche() ;

                ch = toupper(ch) ;

                if (ch == '0')

                        return ;

        }
        while (ch != 'N' && ch != 'Y') ;

        if (ch == 'Y')

        {

                code = tcode ;

                fstream file ;

                file.open("PRODUCT.DAT", ios::out | ios::app ) ;

                file.write((char *) this, sizeof(product)) ;

                file.close() ;

                tcode++ ;

        }
        do

        {

                gotoxy(1,17) ;

                clreol() ;
```

```
                    gotoxy(1,24) ;

                    clreol() ;

                    gotoxy(1,25) ;

                    clreol() ;

                    gotoxy(5,17) ;

                    cout <<"Do you want to add more records (y/n) : "
;

                    ch = getche() ;

                    ch = toupper(ch) ;

                    if (ch == '0')

                            return ;

            }

            while (ch != 'N' && ch != 'Y') ;

        }

        while (ch == 'Y') ;

}



void product :: display_record(int tcode)

{

        fstream file ;

        file.open("PRODUCT.DAT", ios::in) ;

        file.seekg(0,ios::beg) ;

        while (file.read((char *) this, sizeof(product)))

        {

                if (code == tcode)
```

```
                        {
                                gotoxy(5,3) ;

                                cout <<"Code : "<<code ;

                                gotoxy(5,4) ;

                                cout <<"Name : "<<name ;

                                gotoxy(5,5) ;

                                cout <<"Cost : "<<cost ;

                                gotoxy(5,6) ;

                                cout <<"Price : "<<price ;

                                break ;

                        }

                }

        file.close() ;

}



int product :: found(int tcode)

{

        fstream file ;

        file.open("PRODUCT.DAT", ios::in) ;

        file.seekg(0,ios::beg) ;

        int found=0 ;

        while (file.read((char *) this, sizeof(product)))

        {

                if (code == tcode)

                {
```

```cpp
                            found++ ;

                            break ;

                    }

            }

            file.close() ;

            return found ;

}


int product :: recordno(int tcode)

{

        fstream file ;

        file.open("PRODUCT.DAT", ios::in) ;

        file.seekg(0,ios::beg) ;

        int found=0 ;

        while (file.read((char *) this, sizeof(product)))

        {

                found++ ;

                if (code == tcode)

                            break ;

        }

        file.close() ;

        return found ;

}
```

```cpp
void product :: delete_record(int tcode)          //DELETE delete()  hehe
{
        fstream file ;
        file.open("PRODUCT.DAT", ios::in) ;
        fstream temp ;
        temp.open("temp.dat", ios::out) ;
        file.seekg(0,ios::beg) ;
        while ( !file.eof() )
        {
                file.read((char *) this, sizeof(product)) ;
                if ( file.eof() )
                        break ;
                if ( code != tcode )
                        temp.write((char *) this, sizeof(product)) ;
        }
        file.close() ;
        temp.close() ;
        file.open("PRODUCT.DAT", ios::out) ;
        temp.open("temp.dat", ios::in) ;
        temp.seekg(0,ios::beg) ;
        while ( !temp.eof() )
        {
                temp.read((char *) this, sizeof(product)) ;
                if ( temp.eof() )
                        break ;
                file.write((char *) this, sizeof(product)) ;
```

```
        }
        file.close() ;
        temp.close() ;
}


void product :: delet()
{
        clrscr() ;
        char t_code[5], ch ;
        int t, tcode ;
        gotoxy(3,25) ;
        cout <<"Press <ENTER> to see the list" ;
        gotoxy(5,3) ;
        cout <<"Enter Code of the WEAPON to be deleted : " ;
        gets(t_code) ;
        t = atoi(t_code) ;
        tcode = t ;
        if (t_code[0] == '0')
                return ;
        if (tcode == 0)
        {
                list() ;
                gotoxy(1,25) ;
                clreol() ;
                gotoxy(3,25) ;
```

```cpp
        cout <<"Press <ENTER> to Exit" ;

        gotoxy(5,24) ;

        cout <<"Enter Code to be deleted : " ;

        gets(t_code) ;

        t = atoi(t_code) ;

        tcode = t ;

        if (tcode == 0)

                   return ;

}

clrscr() ;

if (!found(tcode))

{

        gotoxy(5,5) ;

        cout <<"\n Record not found" ;

        getch() ;

        return ;

}

display_record(tcode) ;

do

{

        gotoxy(1,8) ;

        clreol() ;

        gotoxy(5,8) ;

        cout <<"Do you want to delete this record (Y/N) : " ;

        ch = getche() ;
```

```cpp
                ch = toupper(ch) ;       //so that input is taken as 'Y' or 'N'
regardless of case
        }
        while (ch != 'N' && ch != 'Y') ;
        if (ch == 'N')
                return ;
        delete_record(tcode) ;
        gotoxy(5,15) ;
        cout <<"\n Record Deleted" ;
        getch() ;
}



void product :: insert(int tcode)
{
        int recno ;
        recno = recordno(tcode) ;
        int valid, t_code ;
        char ch, t_cost[10], t_price[10] ;
        gotoxy(71,2) ;
        cout <<"<0>=Exit" ;
        gotoxy(5,12) ;
        cout <<"Code : " ;
        gotoxy(5,14) ;
        cout <<"Name : " ;
        gotoxy(5,16) ;
```

```
cout <<"Cost : " ;

gotoxy(5,18) ;

cout <<"Price : " ;

do

{

        gotoxy(20,12) ;

        clreol() ;

        cout <<"Change (y/n) : " ;

        ch = getche() ;

        ch = toupper(ch) ;

        if (ch == '0')

                        return ;

}

while (ch != 'N' && ch != 'Y') ;

valid = 0 ;

while (ch == 'Y' && !valid)

{

        valid = 1 ;

        gotoxy(1,12) ;

        clreol() ;

        gotoxy(1,24) ;

        clreol() ;

        gotoxy(1,25) ;

        clreol() ;

        gotoxy(3,25) ;

        cout <<"ENTER CODE TO BE ADDED IN THE MENU" ;
```

```
        gotoxy(5,12) ;

        cout <<"Code : " ;

        cin>>t_code;

        if (t_code == 0)

                    return ;

        if (found(t_code) && t_code != tcode)

        {

                    valid = 0 ;

                    gotoxy(3,24) ;

                    cout <<"\n CODE ALREADY GIVEN" ;

                    getch() ;

        }

}

do

{

        gotoxy(20,14) ;

        clreol() ;

        cout <<"Change (Y/N) : " ;

        ch = getche() ;

        ch = toupper(ch) ;

        if (ch == '0')

                    return ;

}

while (ch != 'N' && ch != 'Y') ;

valid = 0 ;

while (ch == 'Y' && !valid)
```

```
{
        valid = 1 ;
        gotoxy(1,14) ;
        clreol() ;
        gotoxy(1,24) ;
        clreol() ;
        gotoxy(1,25) ;
        clreol() ;
        gotoxy(3,25) ;
        cout <<"ENTER ITEM NAME TO ADD IN THE MENU" ;
        gotoxy(5,14) ;
        cout <<"Item Name : " ;
        gets(name) ;
        strupr(name) ;
        if (name[0] == '0')
                return ;
        if ((strlen(name) < 1) || (strlen(name) > 20))
        {
                valid = 0 ;
                gotoxy(3,24) ;
                cout <<"\n Range = 1..20" ;
                getch() ;
        }
}
do
{
```

```
            gotoxy(20,16) ;

            clreol() ;

            cout <<"Change (Y/N) : " ;

            ch = getche() ;

            ch = toupper(ch) ;

            if (ch == '0')

                        return ;

}

while (ch != 'N' && ch != 'Y') ;

valid = 0 ;

while (ch == 'Y' && !valid)

{

            valid = 1 ;

            gotoxy(1,16) ;

            clreol() ;

            gotoxy(1,24) ;

            clreol() ;

            gotoxy(1,25) ;

            clreol() ;

            gotoxy(3,25) ;

            cout <<"ENTER ITEM COST TO ADD IN THE MENU" ;

            gotoxy(5,16) ;

            cout <<"Item Cost : " ;

            gets(t_cost) ;

            cost = atof(t_cost) ;

            if (t_cost[0] == '0')
```

```cpp
                    return ;
        if (cost < 1 || cost > 800)
        {
                    valid = 0 ;
                    gotoxy(3,24) ;
                    cout <<"\n Range = 1..800" ;
                    getch() ;
        }
}
do
{
        gotoxy(20,18) ;
        clreol() ;
        cout <<"Change (Y/N) : " ;
        ch = getche() ;
        ch = toupper(ch) ;
        if (ch == '0')
                    return ;
}
while (ch != 'N' && ch != 'Y') ;
valid = 0 ;
while (ch == 'Y' && !valid)
{
        valid = 1 ;
        gotoxy(1,18) ;
        clreol() ;
```

```
        gotoxy(1,24) ;

        clreol() ;

        gotoxy(1,25) ;

        clreol() ;

        gotoxy(3,25) ;

        cout <<"ENTER PRICE TO ADD IN THE MENU" ;

        gotoxy(5,18) ;

        cout <<"Price : " ;

        gets(t_price) ;

        price = atof(t_price) ;

        if (t_price[0] == '0')

                return ;

        if (price < cost || price > 1000)

        {

                valid = 0 ;

                gotoxy(3,24) ;

                cout <<"\n Range = " <<cost <<"..1000" ;

                getch() ;

        }

}

do

{

        gotoxy(1,21) ;

        clreol() ;

        gotoxy(1,24) ;

        clreol() ;
```

```
        gotoxy(1,25) ;

        clreol() ;

        gotoxy(5,21) ;

        cout <<"Do you want to save this record (Y/N) : " ;

        ch = getche() ;

        ch = toupper(ch) ;

        if (ch == '0')

                    return ;

}

while (ch != 'N' && ch != 'Y') ;

if (ch == 'N')

        return ;

code = t_code ;

cout <<"\n" <<name ;

cout <<cost ;

cout <<price ;

getch() ;

fstream file ;

file.open("PRODUCT.DAT", ios::out | ios::ate) ;

int location ;

location = (recno-1) * sizeof(product) ;

file.seekp(location) ;

file.write((char *) this, sizeof(product)) ;

file.close() ;

sort() ;

clrscr() ;
```

```cpp
        gotoxy(5,15) ;
        cout <<"\n Record Modified" ;
        getch() ;
}


void product :: modify()
{
        clrscr() ;
        char t_code[5], ch ;
        int t, tcode ;
        gotoxy(3,25) ;
        cout <<"Press <ENTER> to see the list" ;
        gotoxy(5,3) ;
        cout <<"N O P E" ;
        gets(t_code) ;
        t = atoi(t_code) ;
        tcode = t ;
        if (t_code[0] == '0')
                return ;
        if (tcode == 0)
        {
                list() ;                //check function definition in the beginning
                gotoxy(1,25) ;
                clreol() ;
                gotoxy(3,25) ;
```

```cpp
        cout <<"Press <ENTER> to Exit" ;

        gotoxy(5,24) ;

        cout <<"-*_*-" ;

        gets(t_code) ;

        t = atoi(t_code) ;

        tcode = t ;

        if (tcode == 0)

                   return ;

}

clrscr() ;

if (!found(tcode))

{

        gotoxy(5,5) ;

        cout <<"\n Record not found" ;

        getch() ;

        return ;

}

display_record(tcode) ;

do

{

        gotoxy(5,8) ;

        cout <<"NOPE " ;

    //  ch = getche() ;

    //  ch = toupper(ch) ;

}

while (ch != 'N' && ch != 'Y') ;
```

```cpp
        if (ch == 'N')
                return ;
}


void product :: sort()
{
        int i=0,j ;
        product arr[100] ;
        product temp ;
        fstream file ;
        file.open("PRODUCT.DAT", ios::in) ;
        file.seekg(0,ios::beg) ;
        while (file.read((char *) &arr[i], sizeof(product)))
                i++ ;
        int size ;
        size = i ;
        file.close() ;
        for (i=1; i<size; i++)
                for (j=0; j<size-i; j++)
                {
                        if (arr[j].code > arr[j+1].code)
                        {
                                temp=arr[j];
                                arr[j]=arr[j+1];
                                arr[j+1]=temp;
```

```cpp
                        }
                }
        file.open("PRODUCT.DAT", ios::out) ;
        for (i=0; i<size; i++)
                file.write((char *) &arr[i], sizeof(product)) ;
        file.close() ;
}


void product :: buy()
{
        clrscr() ;
        account a ;
        int t_billno, purchased=0 ;
        t_billno = a.last_billno() ;
        t_billno++ ;
        char t_code[5], ch;
        float t_quantity ;
        int t, tcode, i=0, valid ;
        float qty ;
        float t_qty, t_cost, t_price ;
        char t_name[30] ;
        struct date d;
        int d1, m1, y1 ;
        getdate(&d);
        d1 = d.da_day ;
```

```cpp
m1 = d.da_mon ;
y1 = d.da_year ;
do
{
        clrscr() ;
        gotoxy(3,25) ;
        cout <<"Press <ENTER> to see the list" ;
        gotoxy(5,3) ;
        cout <<"Enter Code of the Weapon you wish to Buy : " ;
        gets(t_code) ;
        t = atoi(t_code) ;
        tcode = t ;
        if (t_code[0] == '0')
        {
                if (purchased)
                        a.prepare_bill(t_billno) ;
                return ;
        }
        if (tcode == 0)
        {
                list() ;        //check function definition
                gotoxy(1,25) ;
                clreol() ;
                gotoxy(3,25) ;
                cout <<"Press <ENTER> to Exit" ;
                gotoxy(5,24) ;
```

```cpp
                    cout <<"Enter Code of the Weapon you wish to
Buy : " ;

                    gets(t_code) ;

                    t = atoi(t_code) ;

                    tcode = t ;

                    if (tcode == 0)

                    {

                            if (purchased)

                                    a.prepare_bill(t_billno) ;

                            return ;

                    }

        }

        clrscr() ;

        if (!found(tcode))

        {

                    gotoxy(5,5) ;

                    cout <<"\n Code not found" ;

                    getch() ;

                    if (purchased)

                            a.prepare_bill(t_billno) ;

                    return ;

        }

        gotoxy(60,2) ;

        cout <<"Date:" <<d1 <<"/" <<m1 <<"/" <<y1 ;

        display_record(tcode) ;

        do
```

```cpp
{
        valid = 1 ;
        gotoxy(1,8) ;
        clreol() ;
        gotoxy(1,24) ;
        clreol() ;
        gotoxy(1,25) ;
        clreol() ;
        gotoxy(3,25) ;
        cout <<"ENTER QUANTITY TO BE PURCHASED: " ;

        gotoxy(5,8) ;
        cout <<"Quantity : " ;
        cin>>t_quantity ;
        qty=t_quantity ;
        if (t_quantity == 0)
        {
                if (purchased)
                        a.prepare_bill(t_billno) ;
                return ;
        }
        if (qty < 1 || qty > 800)
        {
                valid = 0 ;
                gotoxy(3,24) ;
                cout <<"\n Range = 1..800" ;
```

```cpp
                        getch() ;
                }
        }
        while (!valid) ;
        do
        {
                gotoxy(5,10) ;
                clreol() ;
                gotoxy(5,10) ;
                cout <<"Do you want to cancel this purchase
(Y/N) : " ;
                ch = getche() ;
                ch = toupper(ch) ;
        }
        while (ch != 'N' && ch != 'Y') ;
        if (ch == 'N')
        {
                purchased = 1 ;
                fstream file ;
                file.open("PRODUCT.DAT", ios::in) ;
                file.seekg(0,ios::beg) ;
                while (file.read((char*) this, sizeof(product)))
                {
                        if (code == tcode)
                        {
                                code = tcode ;
```

```
                                    strcpy(t_name,name) ;
                                    t_cost = cost ;
                                    t_price = price ;
                                    t_qty = qty ;

a.add_bill(t_billno,tcode,t_name,t_qty,t_cost,t_price) ;
                                    i++;
                                    break ;
                            }
                    }
                    file.close() ;
            }
            do
            {
                    gotoxy(5,12) ;
                    clreol() ;
                    gotoxy(5,12) ;
                    cout <<"Do you want to BUY MORE (Y/N) : " ;
                    ch = getche() ;
                    ch = toupper(ch) ;
            }
            while (ch != 'N' && ch != 'Y') ;
        }
        while (ch == 'Y') ;
        a.prepare_bill(t_billno) ;
}
```

```
int account :: last_billno()

{
        fstream file ;

        file.open("BILL.DAT", ios::in) ;

        file.seekg(0,ios::beg) ;

        int t=0 ;

        while (file.read((char *) this, sizeof(account)))

                t = billno ;

        file.close() ;

        return t ;

}



void account :: add_bill(int t_billno, int t_code, char t_name[30], float
t_qty, float t_cost, float t_price)

{
        struct date d;

        int d1, m1, y1 ;

        getdate(&d);

        d1 = d.da_day ;

        m1 = d.da_mon ;

        y1 = d.da_year ;

        dd = d1 ;

        mm = m1 ;
```

```cpp
        yy = y1 ;

        code = t_code ;

        strcpy(name,t_name) ;

        cost = t_cost ;

        price = t_price ;

        quantity = t_qty ;

        billno = t_billno ;

        fstream file ;

        file.open("BILL.DAT", ios::out | ios:: app ) ;

        file.write((char *) this, sizeof(account)) ;

        file.close() ;

}




void account :: prepare_bill(int t_billno)        //BROken . Please FIX ;
update- fixed !

{

        clrscr() ;

        struct date d;

        int d1, m1, y1 ;

        getdate(&d);

        d1 = d.da_day ;

        m1 = d.da_mon ;

        y1 = d.da_year ;

        float total=0.0, total_bill=0.0 ;

        gotoxy(33,3) ;
```

```cpp
cout <<"CUSTOMER BILL" ;

gotoxy(55,5) ;

cout <<"Date:" <<d1 <<"/" <<m1 <<"/" <<y1 ;

gotoxy(8,7) ;

cout <<"Goods PURCHASED" ;

gotoxy(8,8) ;

cout <<"+++++++++++++" ;

gotoxy(8,9) ;

cout <<"Code    Name    Cost    Price    Qty    Total" ;

gotoxy(8,10) ;

cout

<<"=================================================================" ;

int row=11 ;

fstream file ;

file.open("BILL.DAT", ios::in) ;

file.seekg(0) ;

while (file.read((char *) this, sizeof(account)) !=0 )

{

        if (billno == t_billno)

        {

                gotoxy(8,5) ;

                cout <<"BILL NO. # " <<billno ;

                gotoxy(8,6) ;

                cout <<"==============" ;

                gotoxy(10,row) ;

                cout <<code ;
```

```
                    gotoxy(15,row) ;

                    cout <<name ;

                    gotoxy(23,row) ;

                    cout <<cost ;

                    gotoxy(31,row) ;

                    cout <<price ;

                    gotoxy(39,row) ;

                    cout <<quantity ;

                    total=quantity*price ;

                    gotoxy(63,row) ;

                    cout<<total;

                    total_bill+=total ;

                    row++ ;

            }
    }
    file.close() ;

    gotoxy(39,row+1) ;

    cout <<"TOTAL BILL: Rs." <<total_bill <<" /-" ;

    getch() ;
}



void account :: bill_list()     // FIXED SPACING. Called when 4 is
pressed in Main Menu

{
        clrscr() ;                              //Page 1 of receipt working Perf
```

```cpp
fstream file ;

file.open("BILL.DAT", ios::in) ;

file.seekg(0) ;

int row=5, found=0, pageno=1, prev_billno=0, flag=0 ;

float total=0.0, total_bill=0.0 ;

gotoxy(30,2) ;

cout <<"LIST OF BILLS" ;

gotoxy(3,4) ;

cout <<"Billno.  Date    Code  Name  Cost  Price  Qty    Total" ;

gotoxy(3,5) ;

cout<<"========================================================================" ;

while (file.read((char *) this, sizeof(account)))
{
        row++ ;
        //delay(20) ;
        found = 1 ;
        if (prev_billno != billno)
        {
                if (flag)
                {
                        gotoxy(54,row) ;
                        cout <<"TOTAL Bill: Rs." <<total_bill <<"/-" ;

                        total_bill = 0.0 ;
```

```
                        row++ ;
            }
            gotoxy(4,row) ;
            cout <<billno ;
}
flag = 1 ;
gotoxy(11,row) ;
cout <<dd <<"/" <<mm <<"/" <<yy ;
gotoxy(24,row) ;
cout <<code ;
gotoxy(28,row) ;
cout <<name ;
gotoxy(36,row) ;
cout <<cost ;
gotoxy(44,row) ;
cout <<price ;
gotoxy(53,row) ;
cout <<quantity ;
total = quantity * price ;
gotoxy(60,row) ;
cout <<total ;
total_bill = total_bill + total ;
if ( row >= 23 )
{
            row = 5 ;
```

```cpp
                        gotoxy(66,1) ;                              //Printing JUnk
in page 2 of receipt. Update: Fix'd!

                        cout <<"Page no. : " <<pageno ;

                        pageno++ ;

                        gotoxy(1,25) ;

                        cout <<"Press any key to continue..." ;

                        getche() ;   //Registers any keypress

                        clrscr() ;

                        gotoxy(30,2) ;

                        cout <<"LIST OF BILLS" ;

                        gotoxy(3,4) ;

                        cout <<"Billno.   Date   Code  Name   Cost
Price   Qty   Total" ;

                        gotoxy(3,5) ;

                        cout
<<"=====================================================
=========" ;

            }

         prev_billno = billno ;

     }

     row++ ;

     gotoxy(52,row) ;

     cout <<"TOTAL BILL: Rs." <<total_bill <<"/=" ;

     if ( !found )

     {

          gotoxy(5,10) ;

          cout <<"\n Records not found" ;
```

```cpp
        }
        gotoxy(66,1) ;
        cout <<"Page no. : " <<pageno ;
        gotoxy(1,25) ;
        cout <<"Press any key to continue..." ;
        getche() ;
        file.close () ;
}


void main()
{
        clrscr() ;
        menu m ;
        m.main_menu() ;
}
```

OUTPUT


```
-----------------------------------------------------------------

                    A M M U - N A T I O N
                    ---------------------

               1: BUY Weapon

               2: CATALOGUE

               3: EDIT PRODUCTS FILE

               4: PRINT RECEIPT

               0: QUIT


               Enter Your Choice : _


-----------------------------------------------------------------
```

Main menu

```
D:\5354TE~1\PROJEC~1\PROJEC~1\PROJV1_6.EXE                    _ □ ×


    Enter Code of the Weapon you wish to Buy : _
```

```
Press <ENTER> to see the list
```
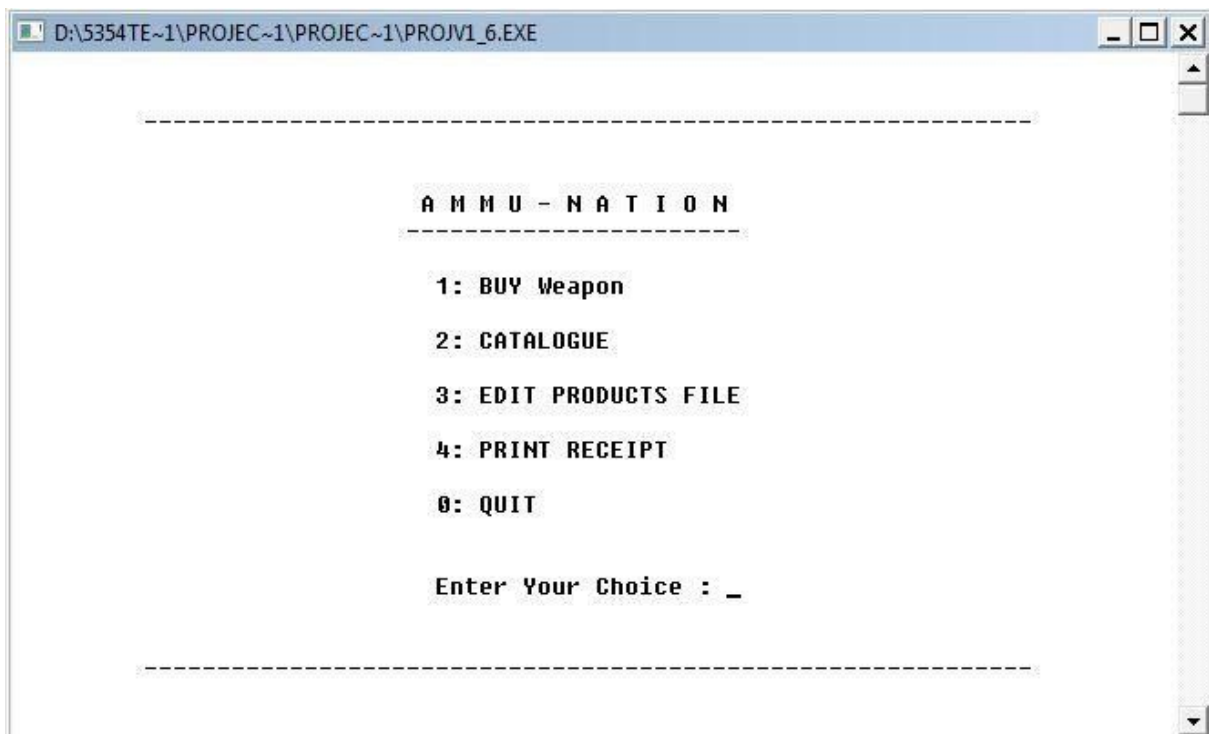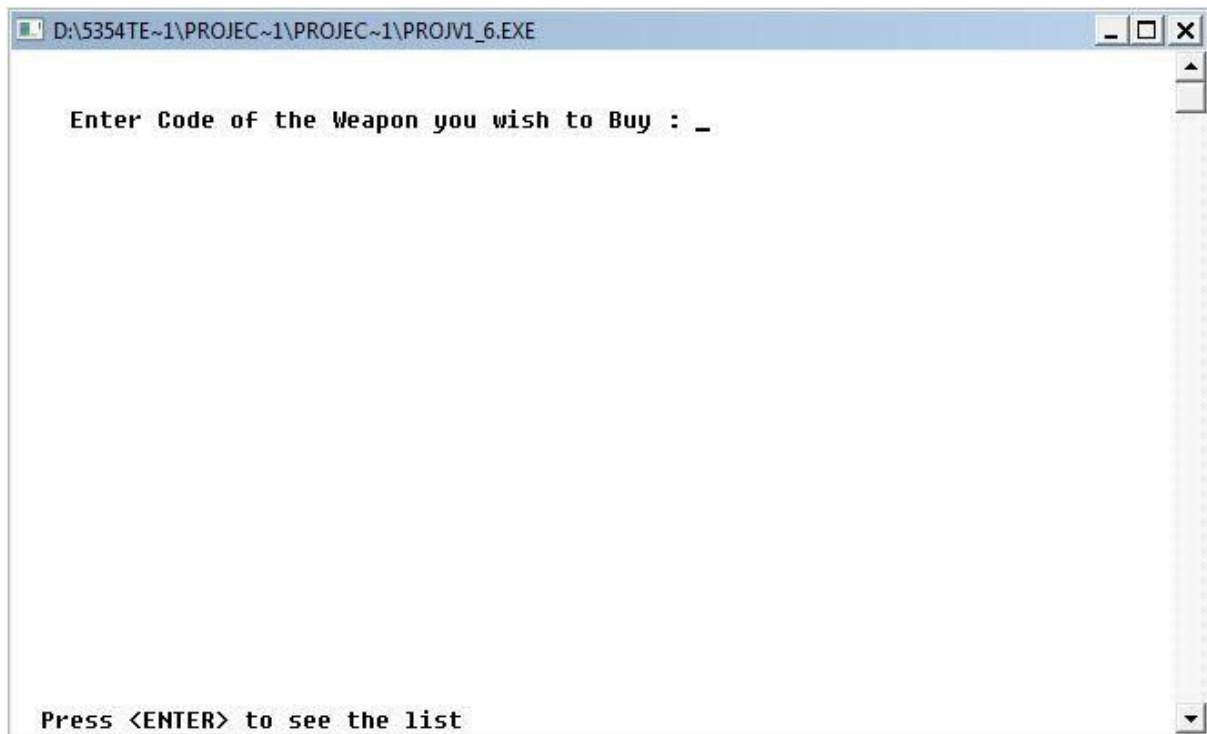
Upon option '1' being selected

```
D:\5354TE~1\PROJEC~1\PROJEC~1\PROJV1_6.EXE                    _ □ ×
                        LIST OF ITEMS

    CODE        NAME            COST            PRICE
    ******************************************************
      1         AK-47           2700            300
      2         AUG             3300            300
      3         AWP             4750            100
      4         C275 AUTO       500             300
      5         DESERT EAGLE    700             300
      6         DUAL BERETTAS   500             300
      7         FAMAS           2250            300
      8         FIVE-SEVEN      500             300
      9         G3SG1           5000            300
```

```
Press any key to continue._
```
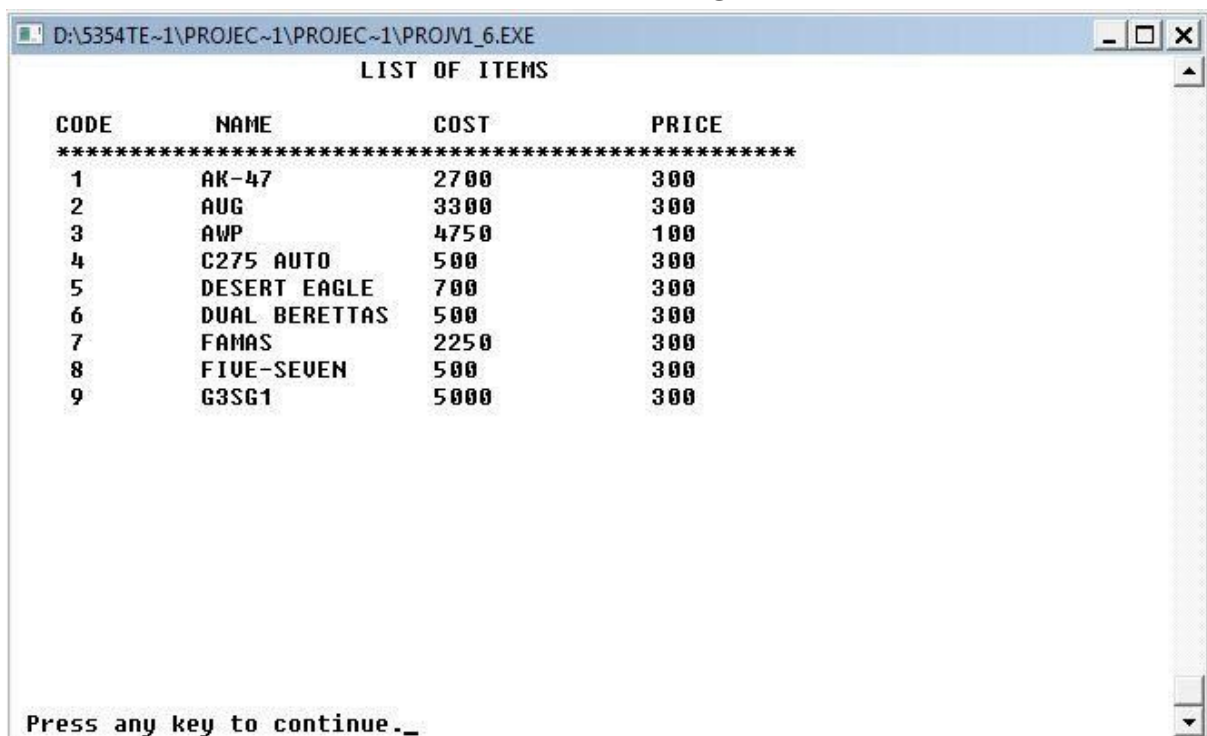
'Enter' key was pressed to access the list of items

```
D:\5354TE~1\PROJEC~1\PROJEC~1\PROJV1_6.EXE
                        LIST OF ITEMS

   CODE          NAME          COST             PRICE
   *****************************************************
     1           AK-47         2700              300
     2           AUG           3300              300
     3           AWP           4750              100
     4           C275 AUTO     500               300
     5           DESERT EAGLE  700               300
     6           DUAL BERETTAS 500               300
     7           FAMAS         2250              300
     8           FIVE-SEVEN    500               300
     9           G3SG1         5000              300




   Enter Code of the Weapon you wish to Buy : _
 Press <ENTER> to Exit
```
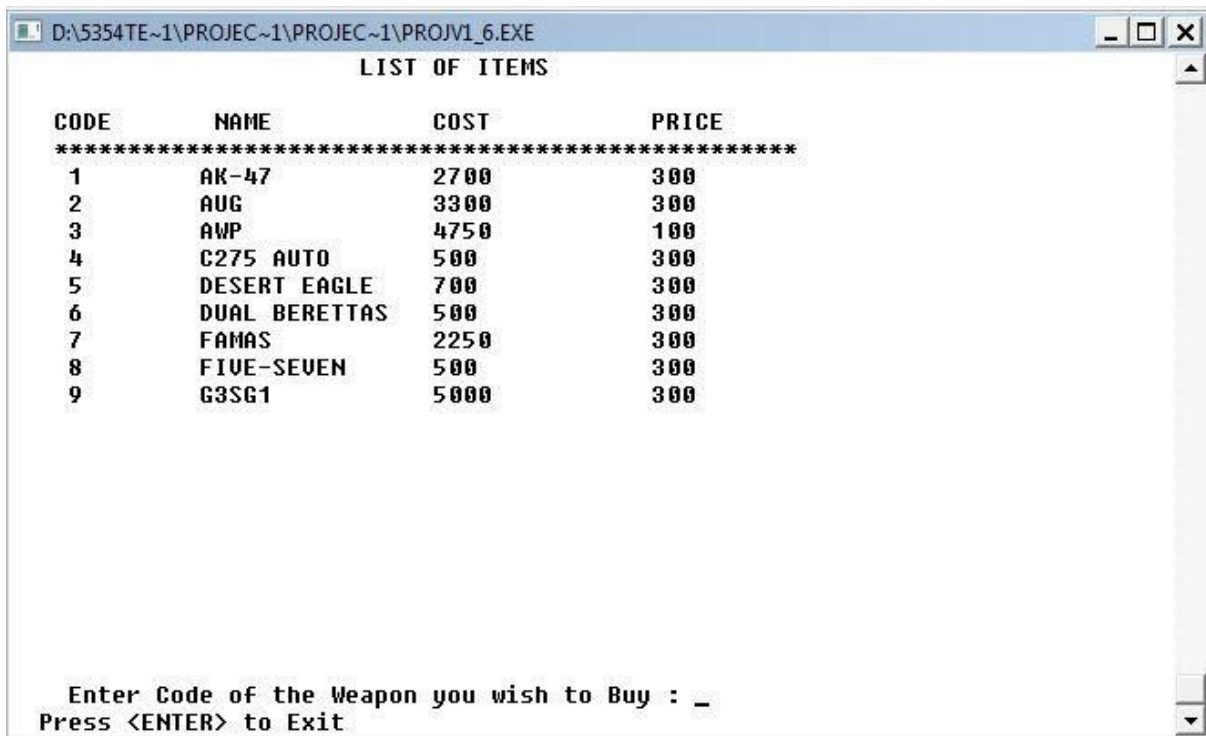
Additional prompt is provided for user to input 'Code'


```
D:\5354TE~1\PROJEC~1\PROJEC~1\PROJV1_6.EXE
                        LIST OF ITEMS

   CODE          NAME          COST             PRICE
   *****************************************************
     1           AK-47         2700              300
     2           AUG           3300              300
     3           AWP           4750              100
     4           C275 AUTO     500               300
     5           DESERT EAGLE  700               300
     6           DUAL BERETTAS 500               300
     7           FAMAS         2250              300
     8           FIVE-SEVEN    500               300
     9           G3SG1         5000              300




   Enter Code of the Weapon you wish to Buy : 1_
 Press <ENTER> to Exit
```

Code '1' is input

```
D:\5354TE~1\PROJEC~1\PROJEC~1\PROJV1_6.EXE                    _ □ ×

                                              Date:16/1/2020
    Code : 1
    Name : AK-47
    Cost : 2700
    Price : 300

    Quantity : _














  ENTER QUANTITY TO BE PURCHASED:
```
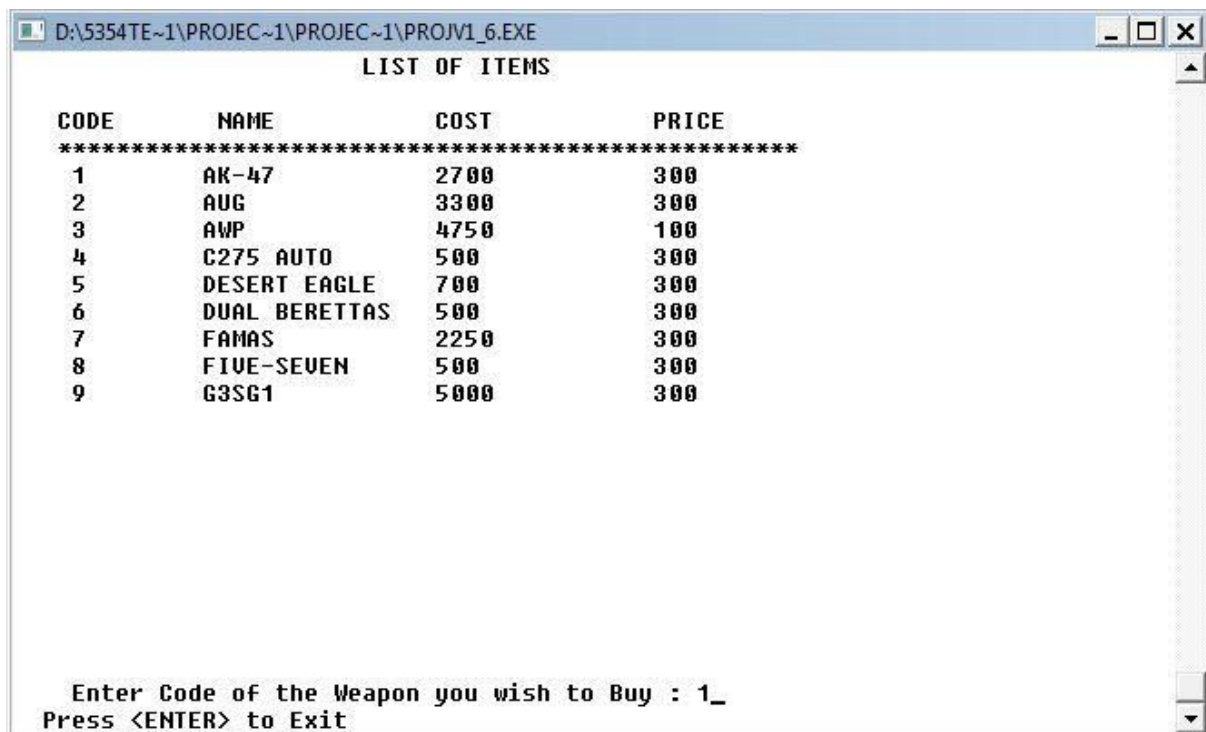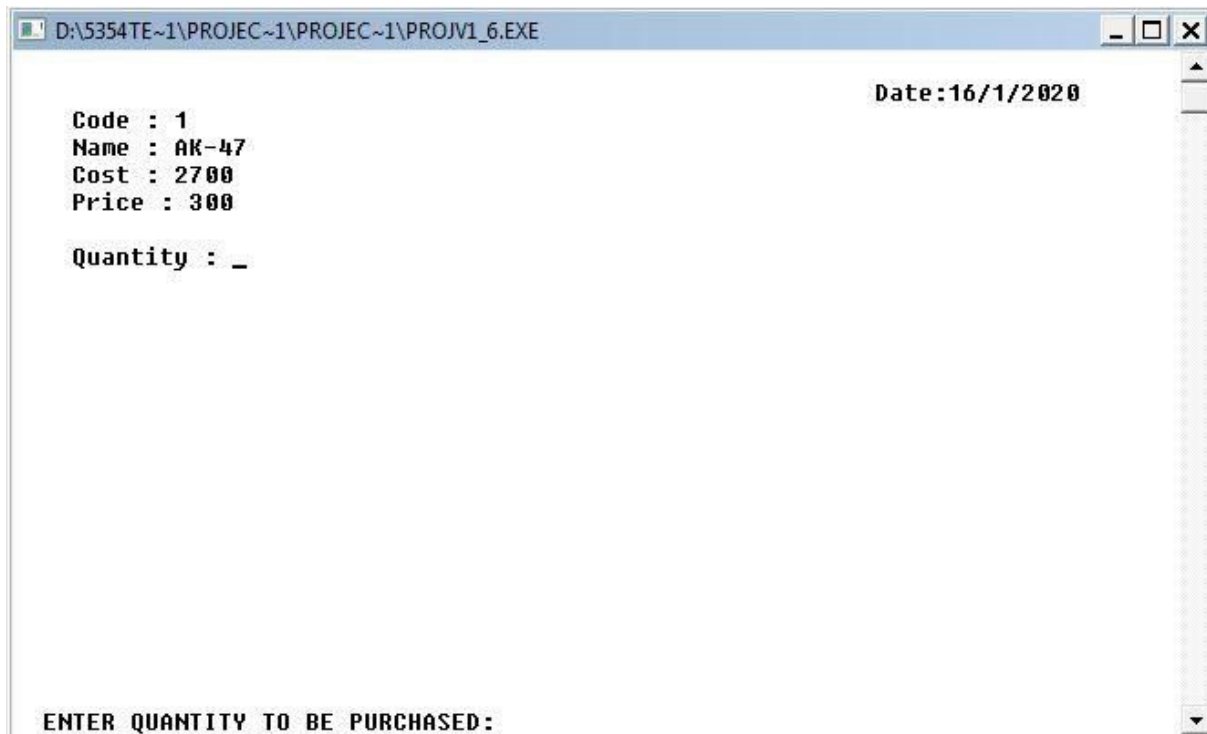
Basic details of weapon with code '1' is displayed with a prompt asking for the required quantity.


```
D:\5354TE~1\PROJEC~1\PROJEC~1\PROJV1_6.EXE                    _ □ ×

                                              Date:16/1/2020
    Code : 1
    Name : AK-47
    Cost : 2700
    Price : 300

    Quantity : 4_














  ENTER QUANTITY TO BE PURCHASED:
```
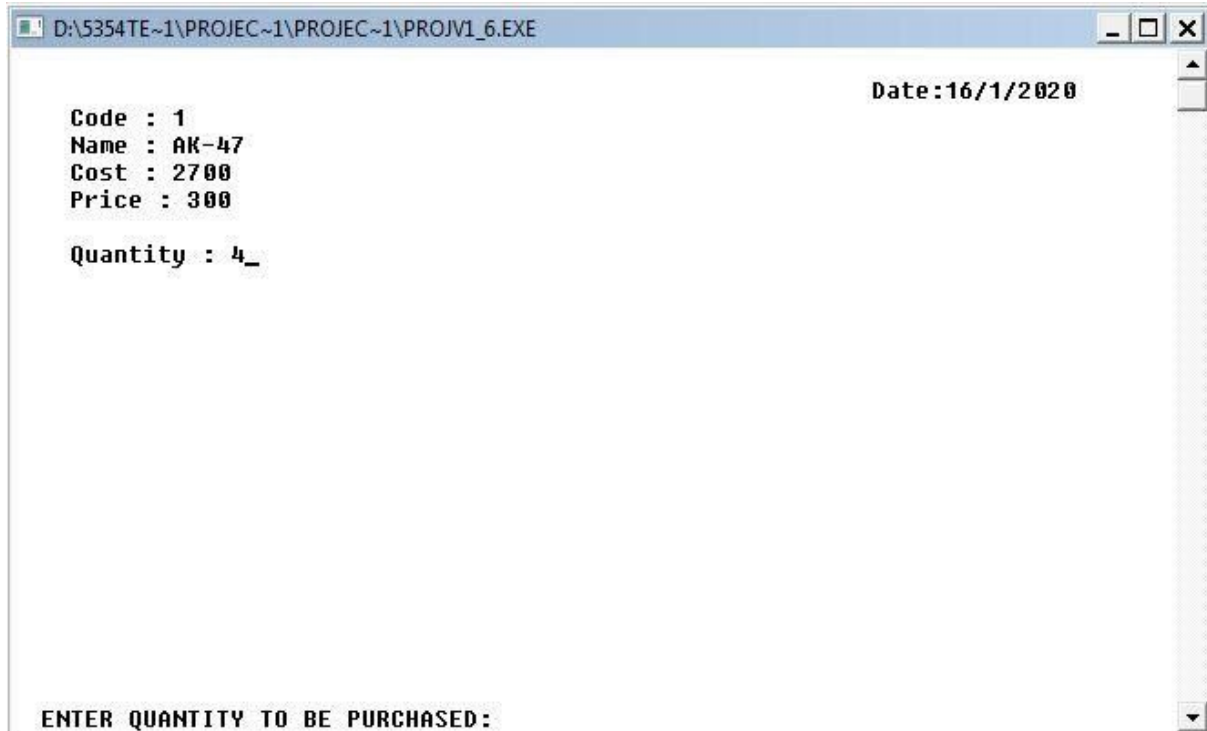
Quantity was specified
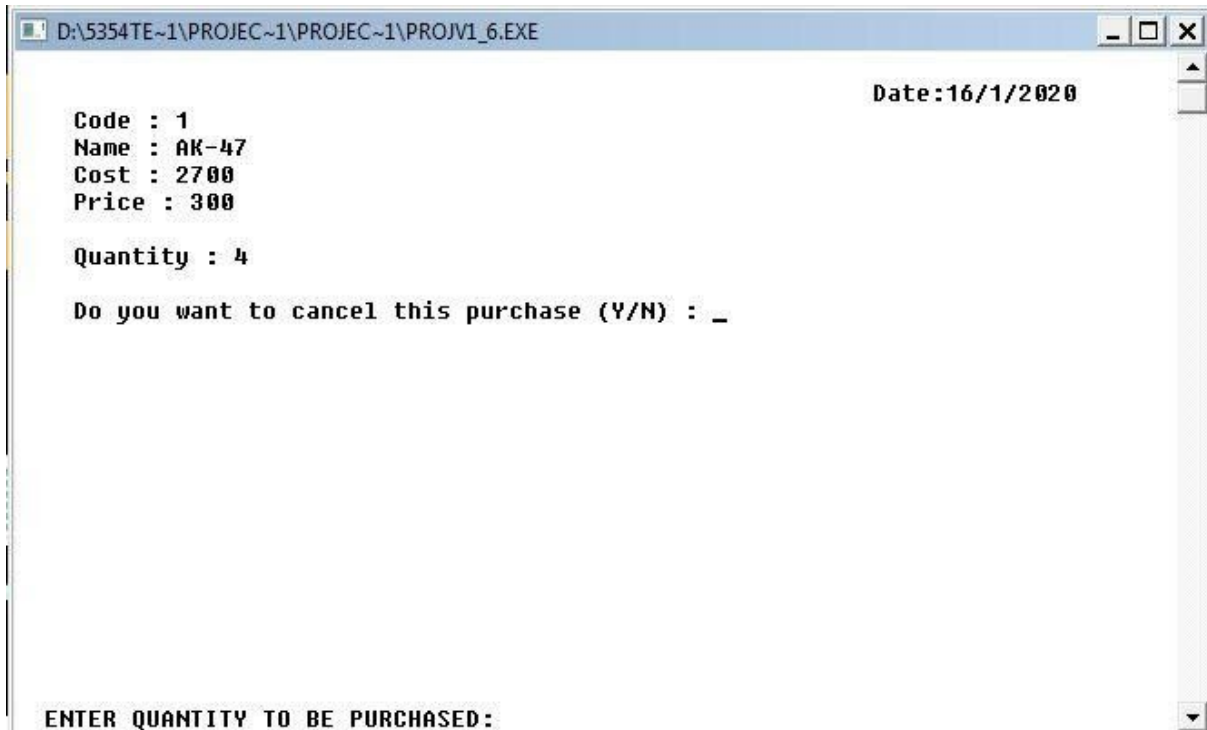
A prompt to confirm the choice pops up.
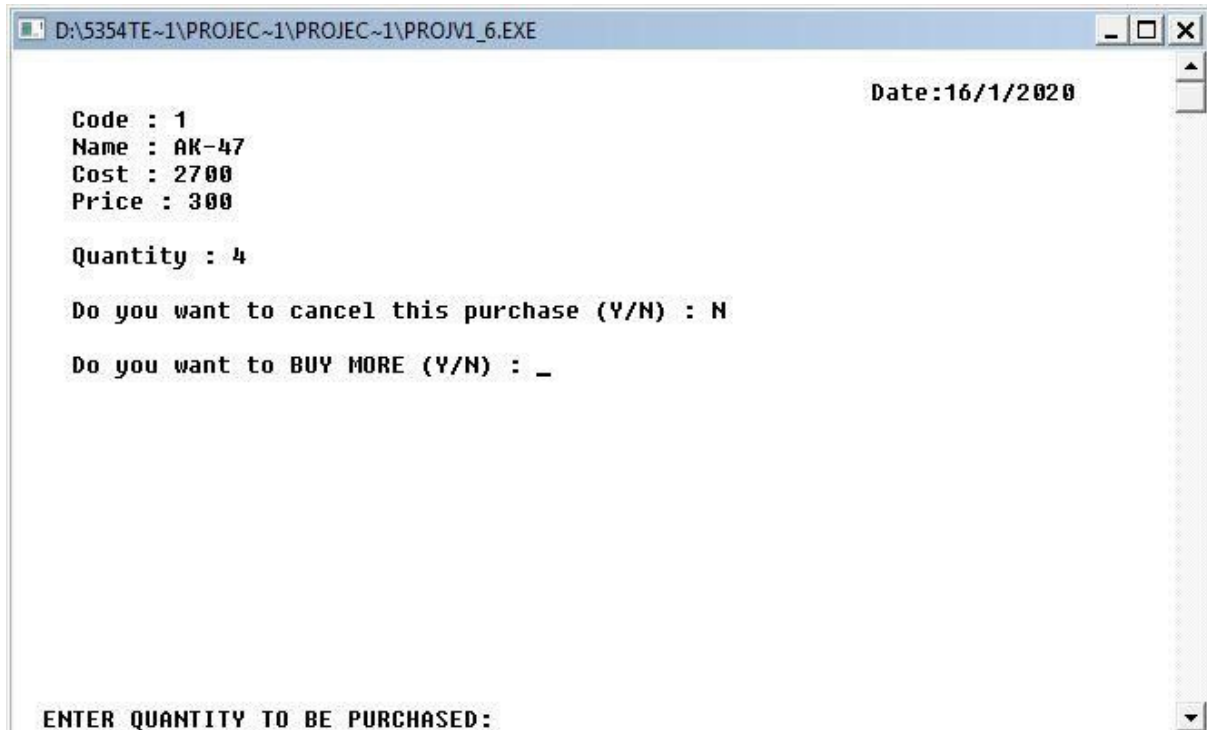
A choice to purchase more weapons in a single session is provided as a prompt.

```
  D:\5354TE~1\PROJEC~1\PROJEC~1\PROJV1_6.EXE                    _ □ ×


      Enter Code of the Weapon you wish to Buy : 2_

























  Press <ENTER> to see the list
```
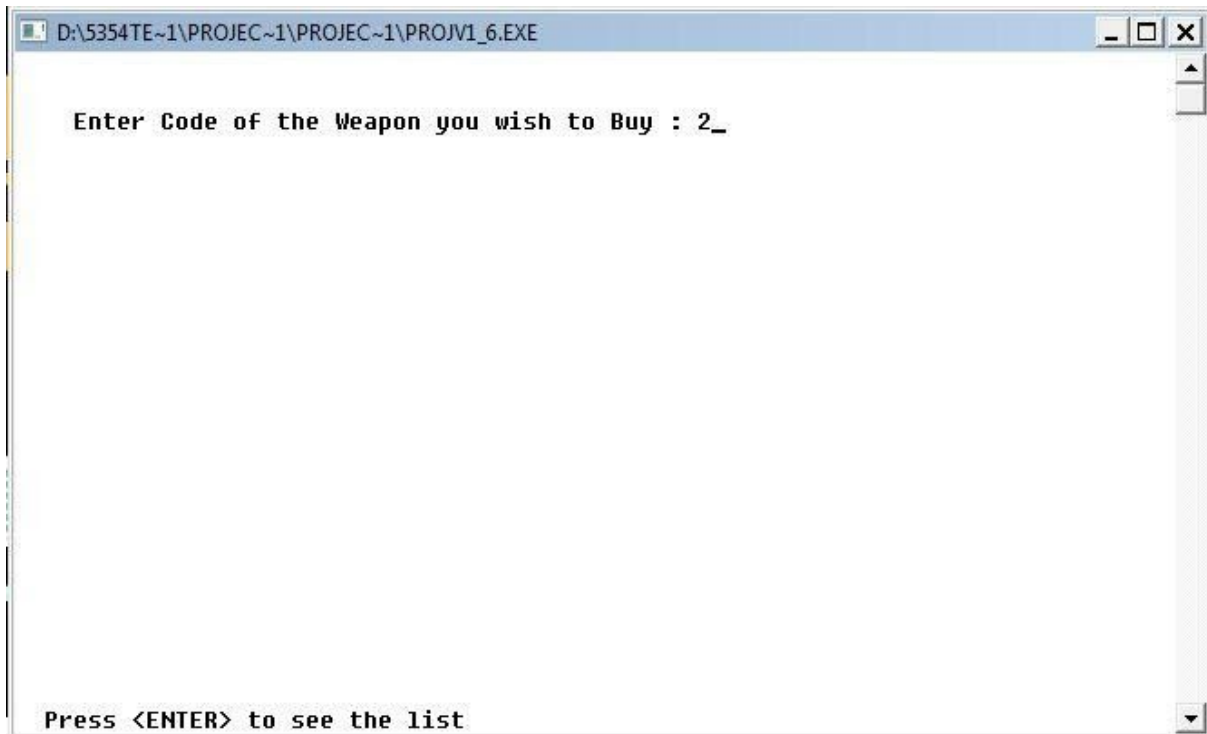
'Y' was chosen thus calling the function again. Code is
asked again.

```
  D:\5354TE~1\PROJEC~1\PROJEC~1\PROJV1_6.EXE                    _ □ ×

                                              Date:16/1/2020
      Code : 2
      Name : AUG
      Cost : 3300
      Price : 300

      Quantity : 2

      Do you want to cancel this purchase (Y/N) : Y

      Do you want to BUY MORE (Y/N) : _










  ENTER QUANTITY TO BE PURCHASED:
```
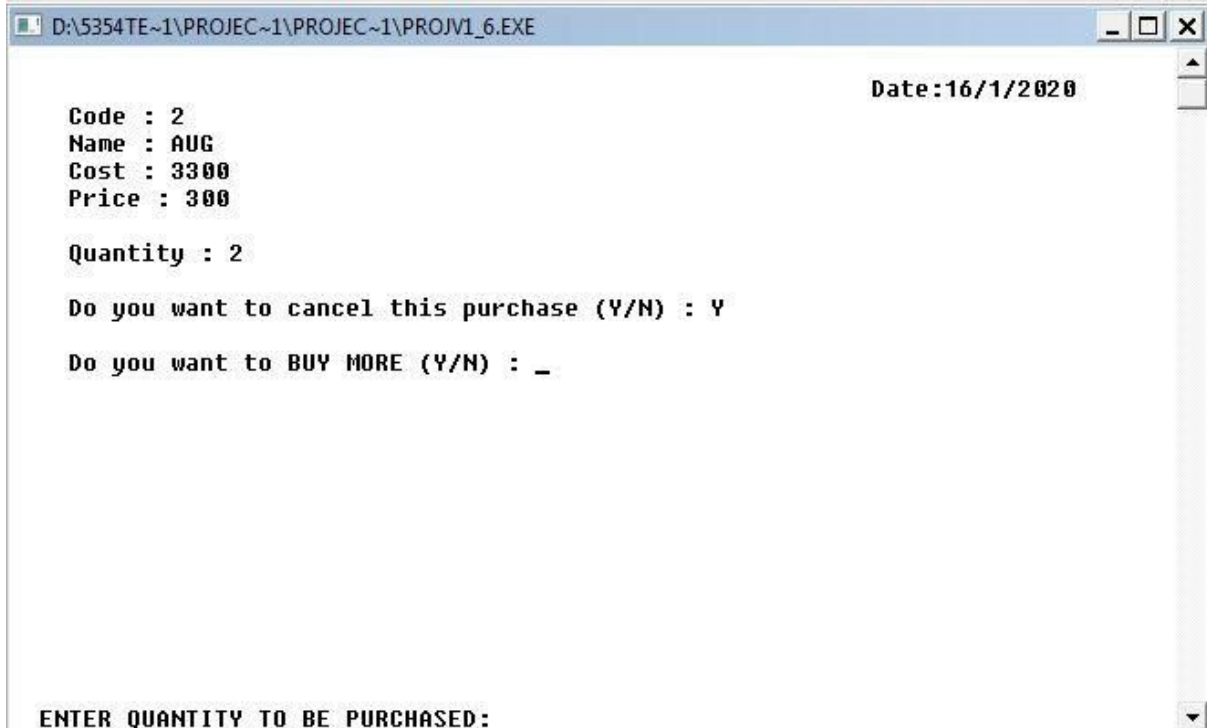
Details of weapon 2 are provided and quantity is asked.
However, the purchase is cancelled.

```
D:\5354TE~1\PROJEC~1\PROJEC~1\PROJV1_6.EXE                              _ □ ×

                         CUSTOMER BILL

       BILL NO. # 8                              Date:16/1/2020
       ==============
       Goods PURCHASED
       ++++++++++++++
       Code    Name    Cost    Price    Qty    Total
       =====================================================================
         1    AK-47    2700     300      4                       1200

                                 TOTAL BILL: Rs.1200 /-_
```
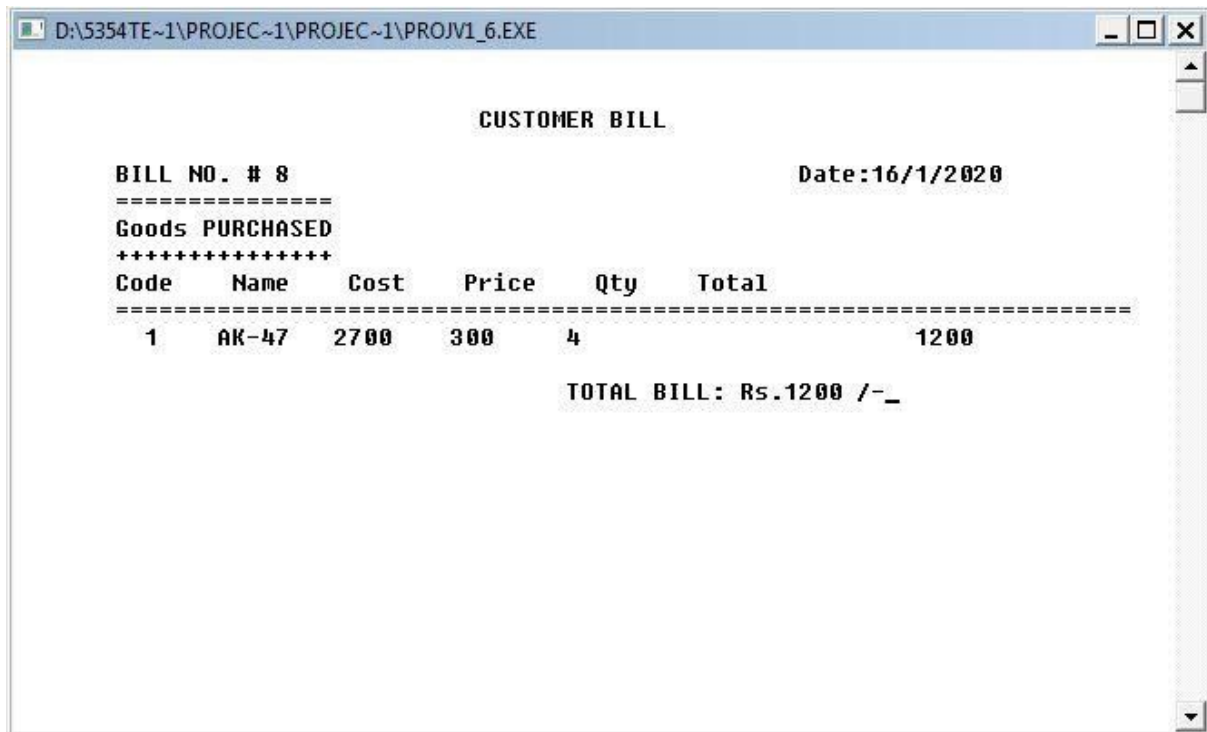
Session ended as 'N' was chosen. Preliminary receipt was printed.

```
D:\5354TE~1\PROJEC~1\PROJEC~1\PROJV1_6.EXE                              _ □ ×

       ----------------------------------------------------------------


                       A M M U - N A T I O N
                       ----------------------

                    1: BUY Weapon

                    2: CATALOGUE

                    3: EDIT PRODUCTS FILE

                    4: PRINT RECEIPT

                    0: QUIT


                    Enter Your Choice : _


       ----------------------------------------------------------------
```
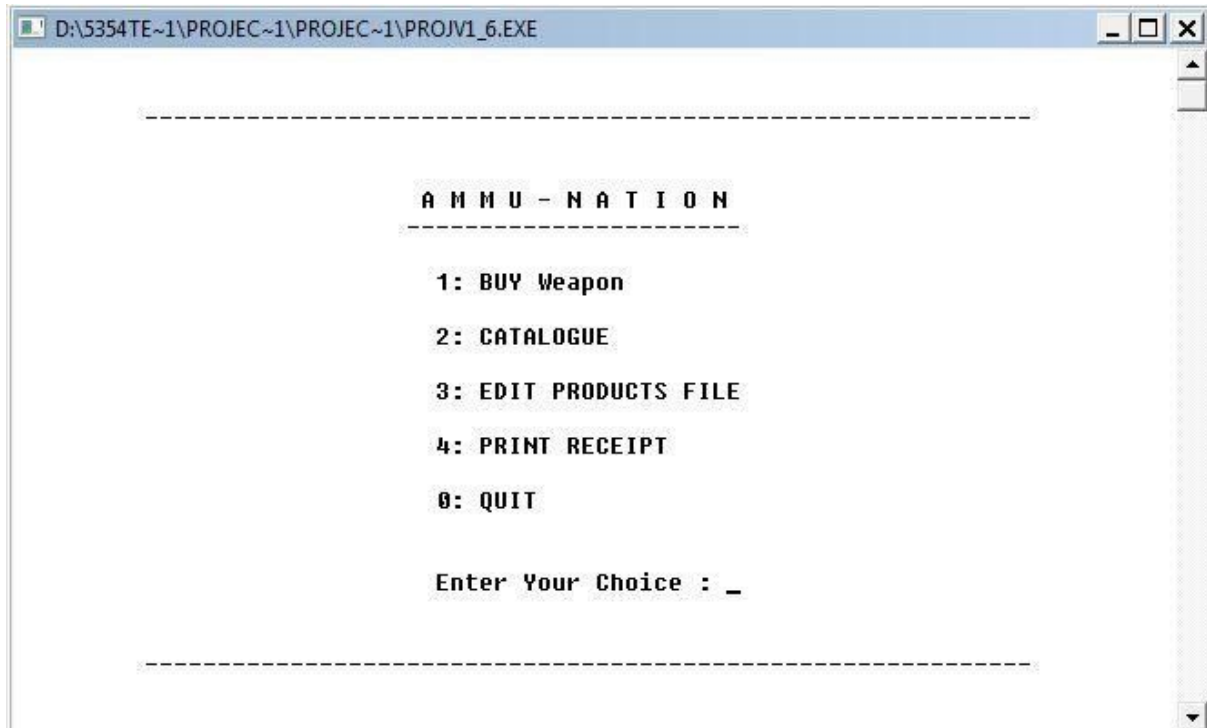
Program reverts back to Main Menu

```
D:\5354TE~1\PROJEC~1\PROJEC~1\PROJV1_6.EXE                    _ □ X
                        LIST OF ITEMS

   CODE         NAME          COST            PRICE
   ***************************************************
    1          AK-47          2700             300
    2          AUG            3300             300
    3          AWP            4750             100
    4          C275 AUTO       500             300
    5          DESERT EAGLE    700             300
    6          DUAL BERETTAS   500             300
    7          FAMAS          2250             300
    8          FIVE-SEVEN      500             300
    9          G3SG1          5000             300




Press any key to continue._
```

Entire list of weapons available are accessed via option '2'

```
D:\5354TE~1\PROJEC~1\PROJEC~1\PROJV1_6.EXE                    _ □ X

       ------------------------------------------------------------


                    A M M U - N A T I O N
                    ---------------------

                    1: BUY Weapon

                    2: CATALOGUE

                    3: EDIT PRODUCTS FILE

                    4: PRINT RECEIPT

                    0: QUIT


                    Enter Your Choice : _


       ------------------------------------------------------------
```

Main Menu is accessed again after pressing any key.

```
D:\5354TE~1\PROJEC~1\PROJEC~1\PROJV1_6.EXE                    _ □ ×
                                                    Page no. : 1
                          LIST OF BILLS

 Billno.   Date        Code  Name        Cost    Price   Qty     Total
 ===============================================================================
   1      19/11/2019    1    AK-47       2700    300     2       600
                                                    TOTAL Bill: Rs.600/-
   2      19/11/2019    2    AUG         3300    300     3       900
          19/11/2019    4    C275 AUT500         300     4       1200
                                                    TOTAL Bill: Rs.2100/-
   3      21/11/2019    9    G3SG1       5000    300     4       1200
                                                    TOTAL Bill: Rs.1200/-
   4      21/11/2019    7    FAMAS       2250    300     7       2100
          21/11/2019    8    FIVE-SEV500         300     5       1500
          21/11/2019    9    G3SG1       5000    300     3       900
                                                    TOTAL Bill: Rs.4500/-
   5      18/12/2019    1    AK-47       2700    300     2       600
                                                    TOTAL Bill: Rs.600/-
   6      18/12/2019    2    AUG         3300    300     3       900
                                                    TOTAL Bill: Rs.900/-
   7      18/12/2019    7    FAMAS       2250    300     3       900
          18/12/2019    5    DESERT E700E        300     6       1800
                                                    TOTAL Bill: Rs.2700/-
   8      16/1/2020     1    AK-47       2700    300     4       1200
 Press any key to continue..._
```

Selecting option '4' displays overall receipt
(cumulative of all sessions)

```
D:\5354TE~1\PROJEC~1\PROJEC~1\PROJV1_6.EXE                    _ □ ×

     ------------------------------------------------------------


                    A M M U - N A T I O N
                    ---------------------

                    1: BUY Weapon

                    2: CATALOGUE

                    3: EDIT PRODUCTS FILE

                    4: PRINT RECEIPT

                    0: QUIT


                    Enter Your Choice : _


     ------------------------------------------------------------
```

Main Menu is accessed after pressing any key.

```
G:\VEDSHE~1\FINALG~1\PROJV1_6.EXE                                    _ □ ×

         ------------------------------------------------------------------

                             --------------------
                              E D I T    M E N U
                             --------------------


                         1. A D D


                         2. D E L E T E


                         0. E X I T

                         Enter Choice :


         ------------------------------------------------------------------
```

Selecting option 3 enables the user to access the add
and delete options.

```
G:\VEDSHE~1\FINALG~1\PROJV1_6.EXE                                    _ □ ×

                                                               <0>=Exit
                              ADDITION
                         --------------------

         Code : 10

         Name : _

         Cost :

         Price :









 ENTER Name TO BE ADDED
```

Selecting option '1' lets the user add a weapon of their choice.


```
 G:\VEDSHE~1\FINALG~1\PROJV1_6.EXE                          _ □ ×

                                                         <0>=Exit
                          ADDITION
                          ---------------------

     Code : 10

     Name : Sickle

     Cost : 2000

     Price : 750_




 ENTER PRICE TO ADD IN THE MENU
```

The basic details required are asked to be input.


```
 G:\VEDSHE~1\FINALG~1\PROJV1_6.EXE                          _ □ ×

                                                         <0>=Exit
                          ADDITION
                          ---------------------

     Code : 10

     Name : Sickle

     Cost : 2000

     Price : 750


     Do you want to save this record (y/n) : y

     Do you want to add more records (y/n) : _
```
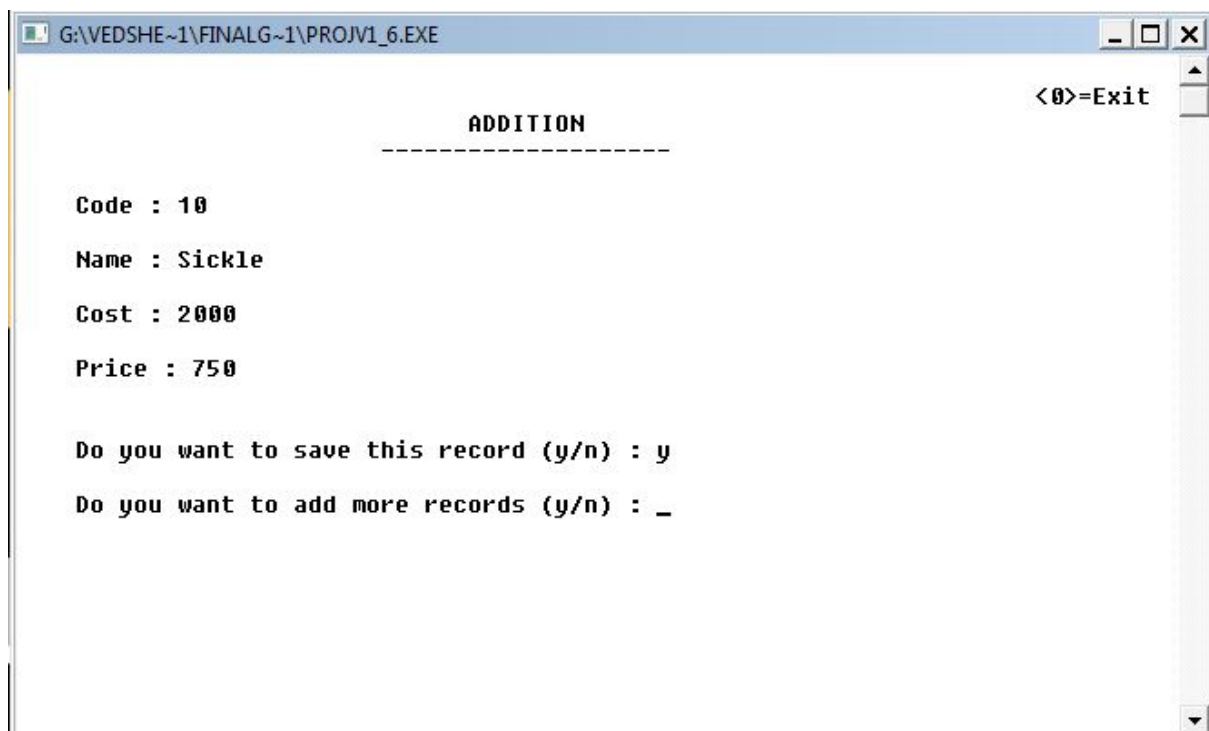
A prompt to confirm the addition appears.

```
 D:\5354TE~1\PROJEC~1\PROJEC~1\PROJV1_6.EXE                              _ □ ×

    ----------------------------------------------------------------


                        A M M U - N A T I O N
                        ----------------------

                    1: BUY Weapon

                    2: CATALOGUE

                    3: EDIT PRODUCTS FILE

                    4: PRINT RECEIPT

                    0: QUIT


                    Enter Your Choice : _


    ----------------------------------------------------------------
```
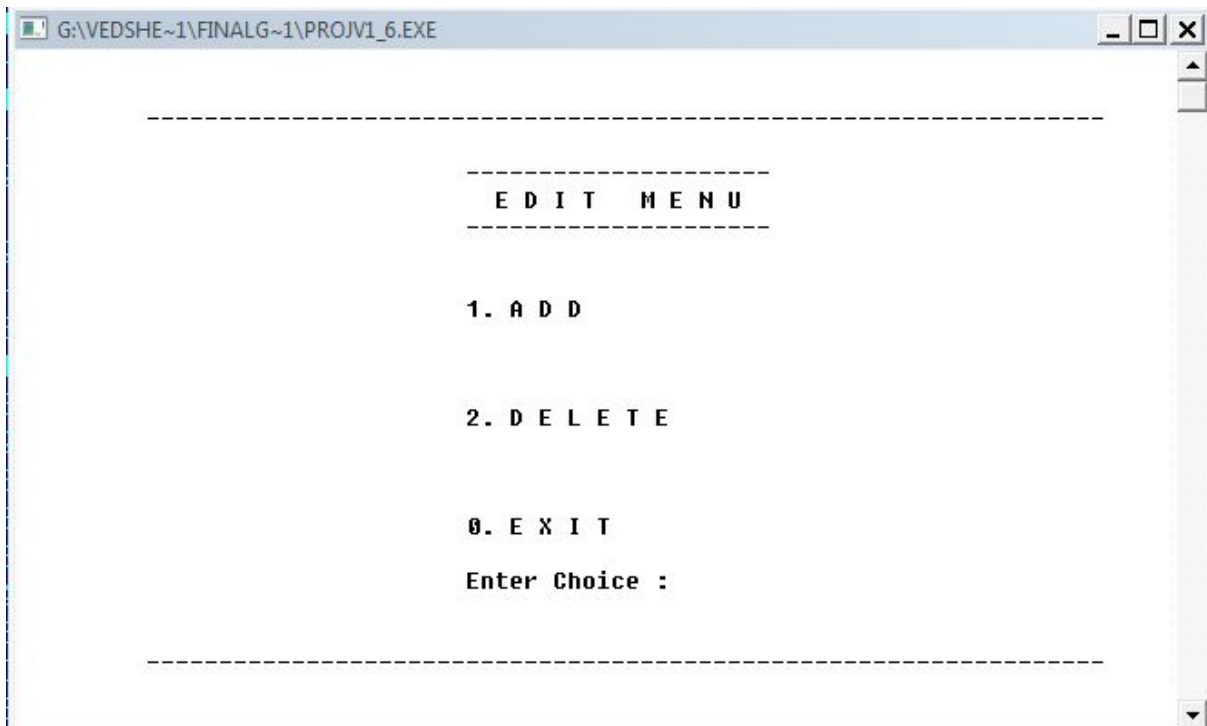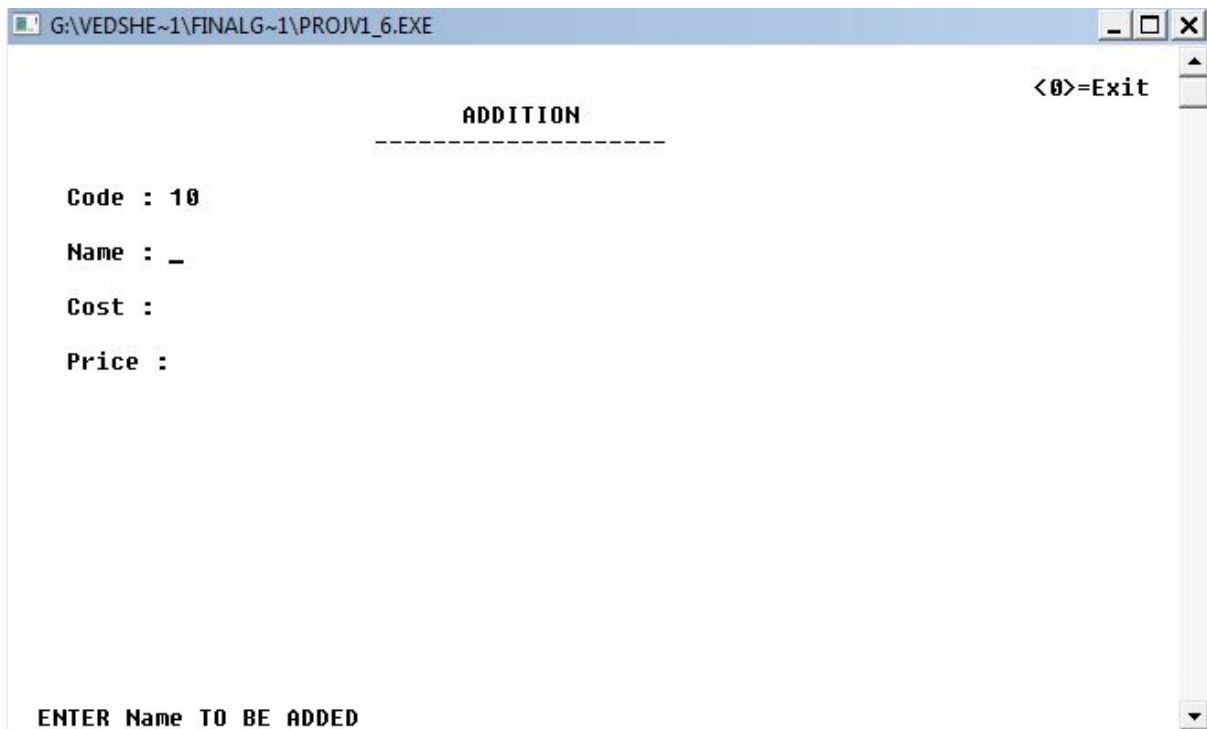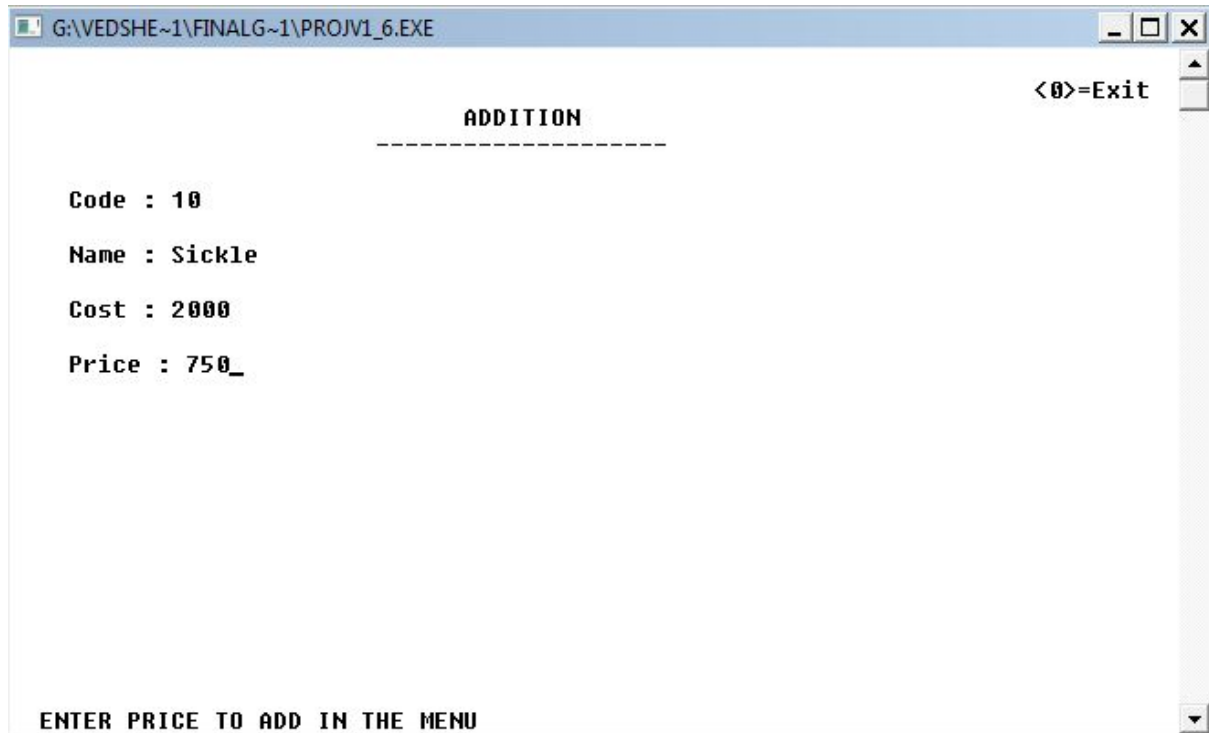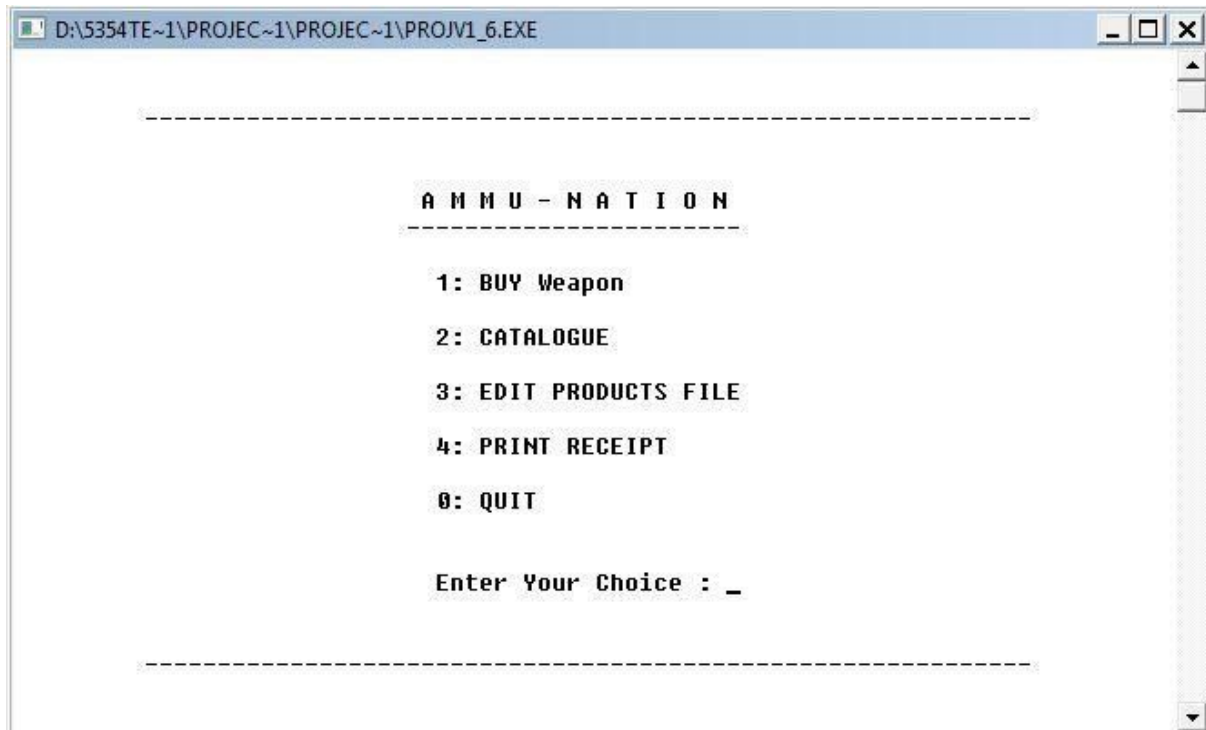
The program reverts back to the Main Menu.

```
 G:\VEDSHE~1\FINALG~1\PROJV1_6.EXE                                       _ □ ×

    ----------------------------------------------------------------

                        ----------------------
                        E D I T   M E N U
                        ----------------------


                    1. A D D



                    2. D E L E T E



                    0. E X I T

                    Enter Choice :


    ----------------------------------------------------------------
```
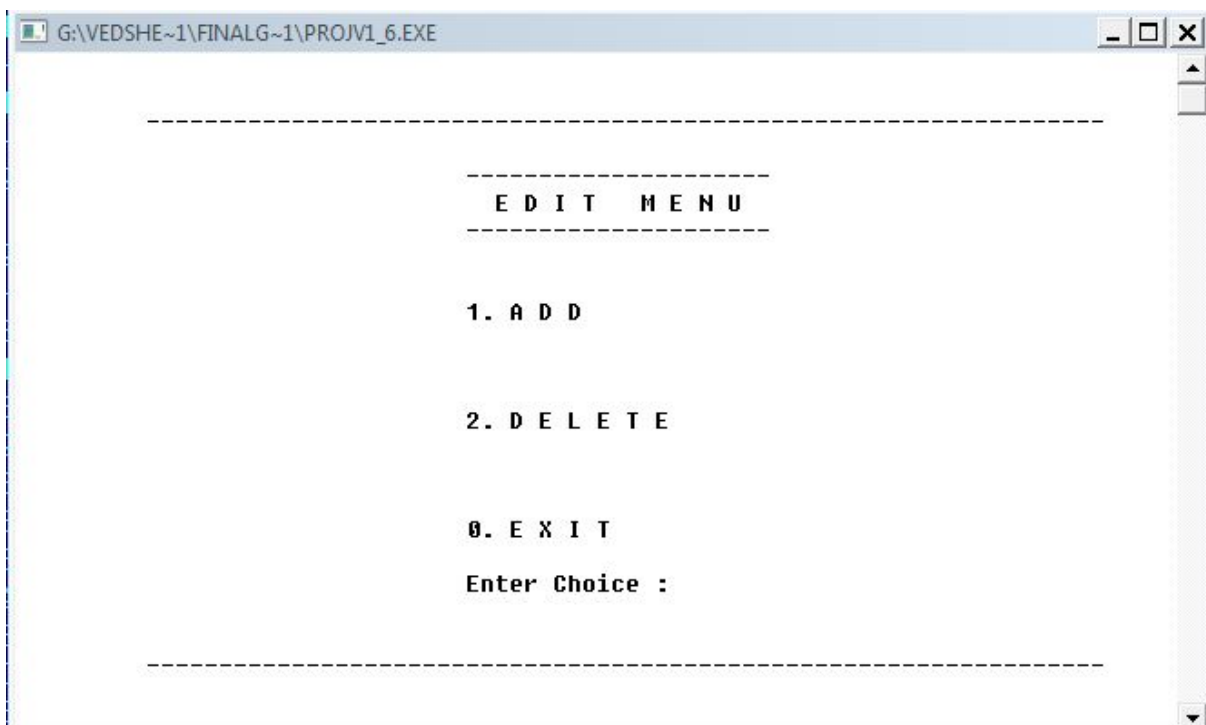
Upon selecting option '3', the user can access the Edit Menu again.

```
■ G:\VEDSHE~1\FINALG~1\PROJV1_6.EXE                                    _ □ ×

    Enter Code of the WEAPON to be deleted :




    Press <ENTER> to see the list
```
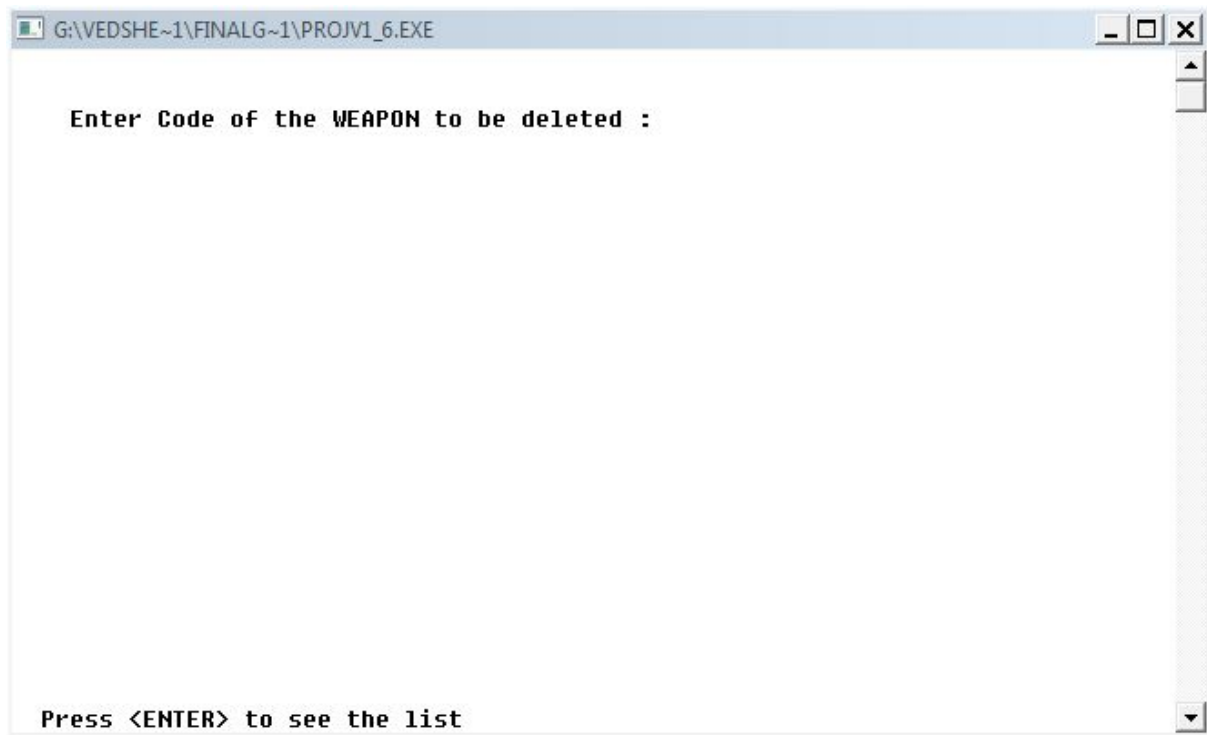
Selecting option '2' lets the user delete any weapon.

```
■ G:\VEDSHE~1\FINALG~1\PROJV1_6.EXE                                    _ □ ×

    Enter Code of the WEAPON to be deleted : 10_




    Press <ENTER> to see the list
```
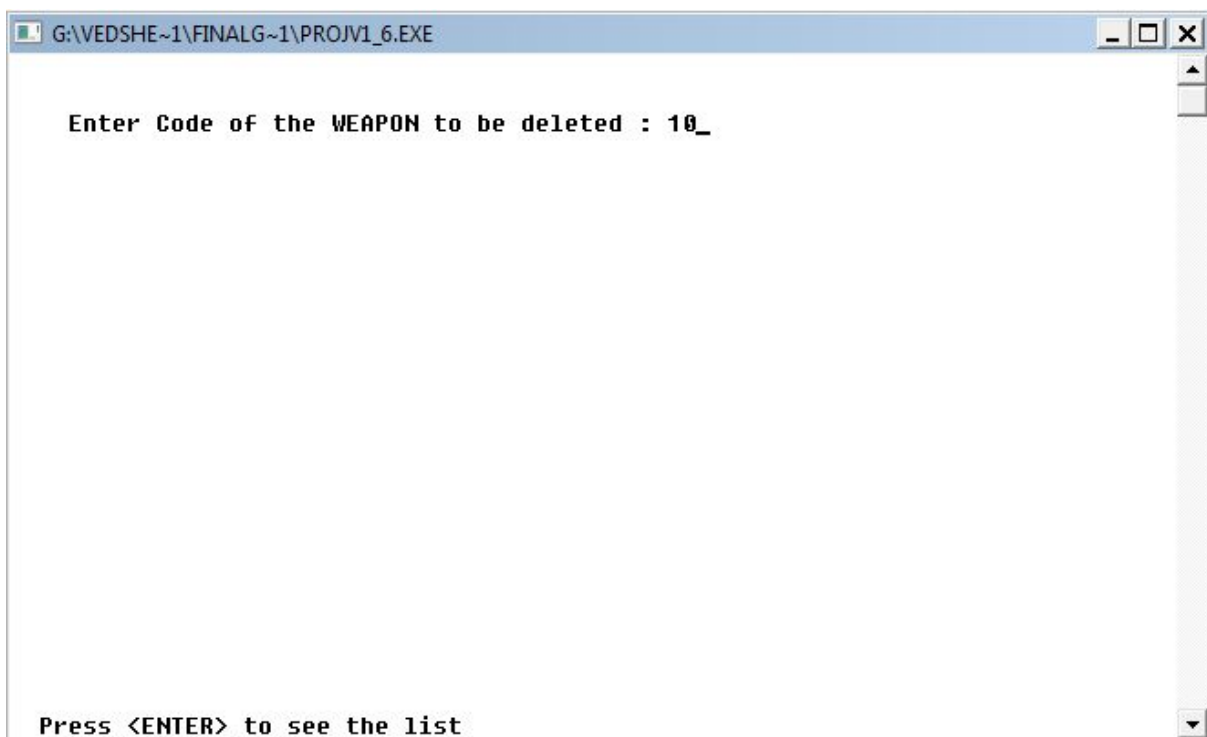
Upon entering the code, the details of the corresponding weapon are displayed.


```
Code : 10
Name : SICKLE
Cost : 2000
Price : 750

Do you want to delete this record (Y/N) : y




Record Deleted
```
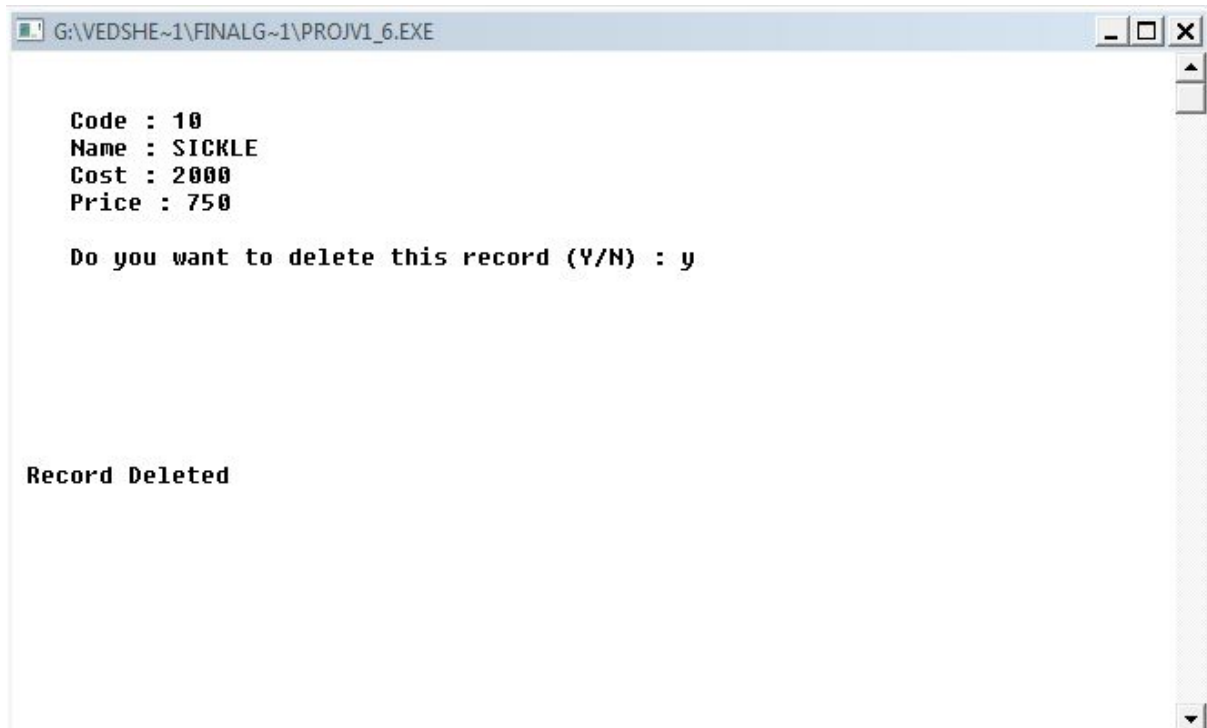
A final prompt appears before confirming the deletion.


```
----------------------------------------------------------------


                    A M M U - N A T I O N
                    ---------------------

                  1: BUY Weapon

                  2: CATALOGUE

                  3: EDIT PRODUCTS FILE

                  4: PRINT RECEIPT

                  0: QUIT


                  Enter Your Choice : _


----------------------------------------------------------------
```

The Main Menu is accessed again.

```
────────────────────────────────────────────────────────────

                    A M M U - N A T I O N
                    ─────────────────────

              1: BUY Weapon

              2: CATALOGUE

              3: EDIT PRODUCTS FILE

              4: PRINT RECEIPT

              0: QUIT


              Enter Your Choice : 0


────────────────────────────────────────────────────────────
```
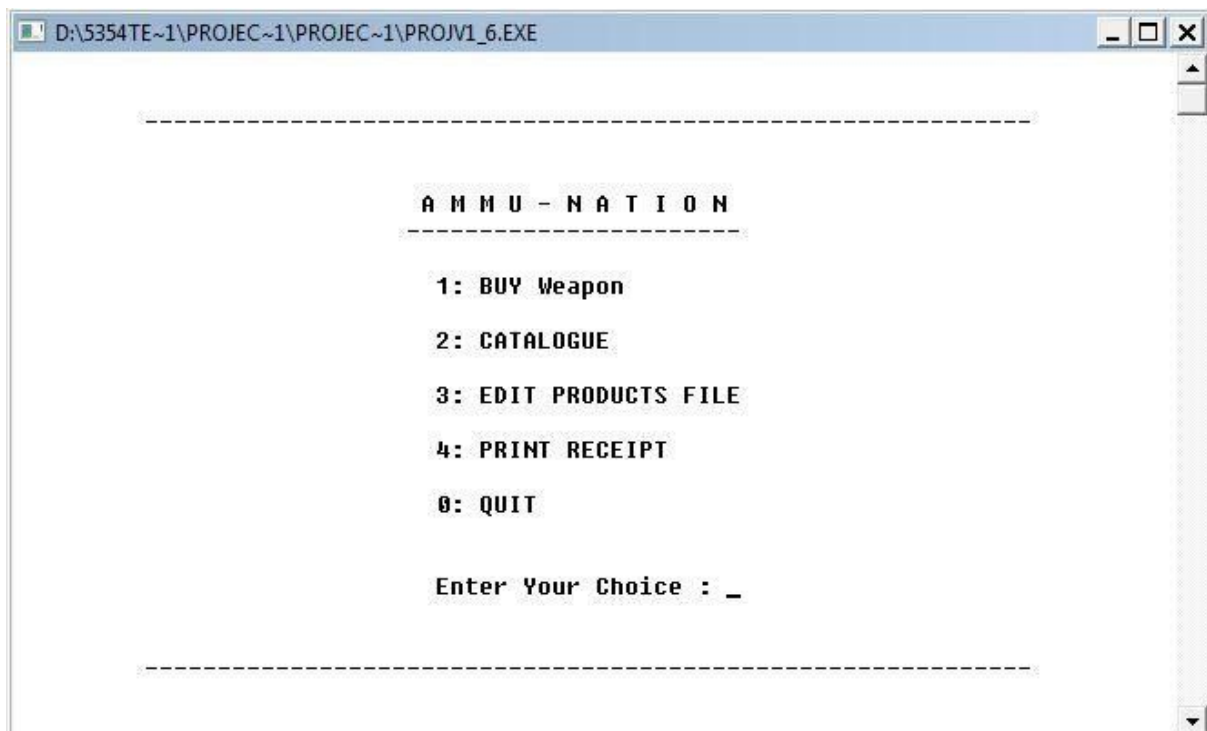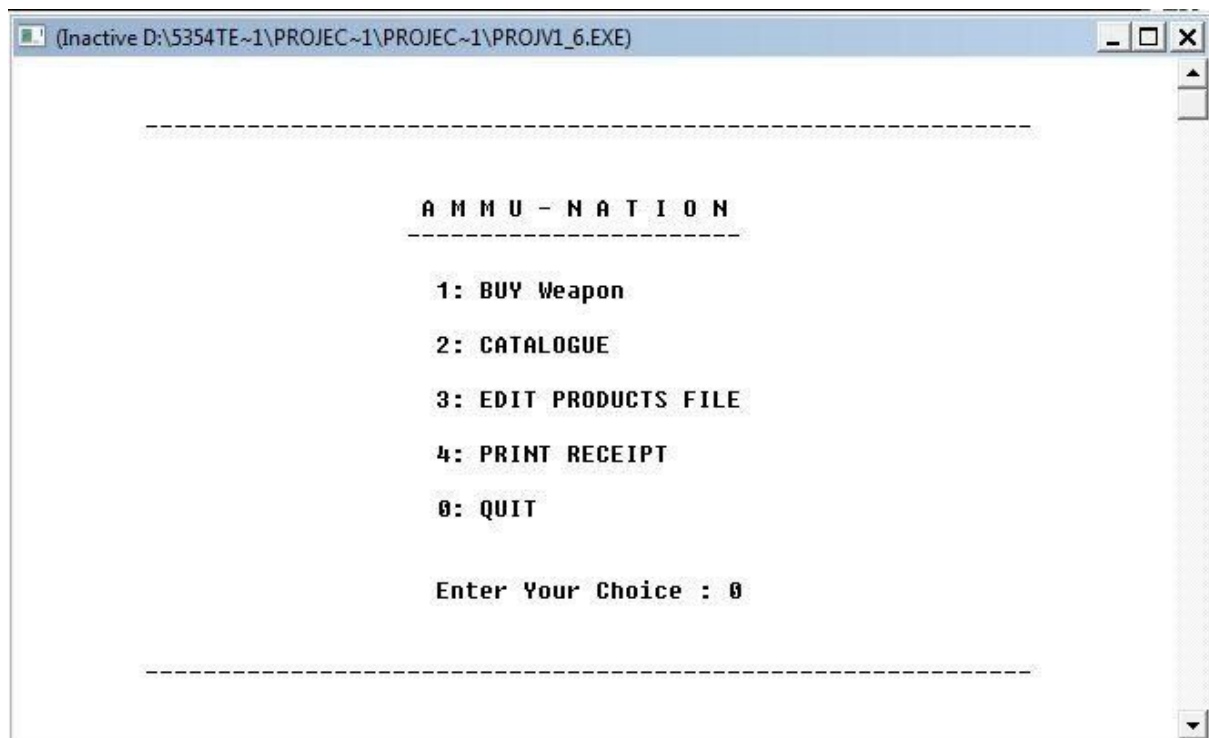
Selecting option '0' renders the output window inactive,
thus exiting the program.

REFERENCES

Sumita Arora Class 12 C++ Textbook