|  |
| --- |
| BQC.jpgKaun-Banega-Crorepati-2011.jpg20121020_stp505.jpg |
| ? ? Quiz ? ? |
| |  |  |  | | --- | --- | --- | | Dhruv Mongia | Roll No : 15 | C++ | |

hgmnmnh

I am grateful to Almighty for giving me the strength to successfully complete my project titled “QUIZ”. I am deeply indebted to *Mrs. Kamaljeet Kaur*, our computer faculty without whose constructive guidance this project would not have been a success. Her valuable advice and suggestions for the corrections, modifications and improvement did enhance the perfection in performing my job. I am grateful to My Parents whose blessings and wishes have gone a long way in the completion of this arduous task. Last but not the least I thank all My Friends and Batch Mates, without their prompt support my efforts would have been in vain.

Table of Contents

Table of Contents

* Introduction
* Hardware & Software requirements
* Components of the program
* Source Code
* Bibliography

Introduction

This program is a great way to create exercises,quizzes and tests and to analyze our performance in them. Unlimited quizzes, unlimited responses! The perfect tool for the classroom, colleagues or just for fun. Create quick and fun quizzes to test their knowledge.

* Features
* Created with the “Keep it simple” principle in mind, it’s a fast way to create your own exercises , quizzes and tests.
* Four different categories to choose from: Maths , Computer , Science & General.
* The user can add, delete or modify endless number of questions and their solutions.
* Adding, deleting or modifying of the questions or their corresponding answers can only be done by the administrator as it is password protected.
* The user can review his performance as his/her score & percentage is displayed at the end of the quiz.
* The source code could be easily modified as it is well complimented by comments.
* Applications

Needless to say, quizzes have made places in our hearts, be it knowledge or entertainment.



The show “KaunBanegaCrorepati” and the famous “Bournvita Quiz” is enjoyed by millions of people.



Quizzes are the best way to know if we have the “Illusion of knowledge” or the true knowledge itself.

Quizzes also remain the best way to screen the smartest people among huge groups.

Components

* Classes & Functions

Classes

1.Class name: menu

Details: It controls over all functioning of project

2.Class name: quiz

Details: It controls over all the functions related to quiz.

Functions

1.Function name: start

Details: It creates the opening screen.

2.Function name: main\_menu

Details: It creates the main menu to control

all the functions

3.Function name: password

Details: It checkswhether the user hasauthorized access or not.

4.Function name: sub\_menu

Details: It creates the sub main menu andreturn file name.

5.Function name: admin\_menu

Details: It creates the admin main menufor adding,deleting and modifying questions.

6.Function name: record\_count

Details: It returns the no. of record in thegiven file.

7.Function name: display\_record

Details: It displays the record for the given

question no. from the given file.

8.Function name: found\_record

Details: It searches the given records in thegiven file and return 0 if not found.

9.Function name: deletion

Details: It gives the question no. to deleterecord from the given file.

10.Function name: modify\_record

Details: It modifies the record for the given

question no. from the given file.

11.Function name: modify

Details: It gives the question no. to modify

record from the given file.

12.Function name: display\_score

Details: It displays the score for the givenplayer's name.

13.Function name: play

Details: It creates screen for playing quizto the player for the given file nameand keep score.

14.Function name: main

Details: It calls the start function andmenu function.

Source Code

#include <iostream.h>

#include <conio.h>

#include <string.h>

#include <fstream.h>

#include <process.h>

#include <stdlib.h>

#include <stdio.h>

#include <ctype.h>

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// CLASS NAME : menu

// DETAILS : IT CONTROLS OVER ALL FUNCTIONING OF PROJECT

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class menu

{

public :

voidmain\_menu(void) ;

char \*sub\_menu(void) ;

void start(void) ;

private :

void password();

voidadmin\_menu(void) ;

} ;

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// CLASS NAME : quiz

// DETAILS : IT CONTROLS OVER ALL THE FUNCTIONS

// RELATED TO QUIZ

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class quiz

{

public :

void play(char filename[13]) ;

void add(char filename[13]) ;

void deletion(char filename[13]) ;

void modify(char filename[13]) ;

private :

intautogen(char filename[13]);

intretqno();

voiddelete\_record(char filename[13], int) ;

voidmodify\_record(char filename[13], int) ;

voiddisplay\_score(char name[20], int, int) ;

intrecordno(char filename[13], int) ;

intrecord\_count(char filename[13]) ;

intfound\_record(char filename[13], int) ;

voiddisplay\_record(char filename[13], int) ;

intques\_no ;

charques[60] ;

char ans1[50] ;

char ans2[50] ;

char ans3[50] ;

char sol ;

} ;

longsi=sizeof(quiz);

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// CLASS NAME : menu

// FUNCTION NAME : start

// DETAILS : IT CREATES THE OPENING SCREEN

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void menu::start()

{ cout<<"\nCOMPUTER QUIZ";

cout<<"\nMADE BY:-";

cout<<"\nDHRUV MONGIA";

cout<<"\nSARTHAK BABBAR";

cout<<"\nAKHIL SHARMA";

getch();

main\_menu();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// CLASS NAME : menu

// FUNCTION NAME :main\_menu

// DETAILS : IT CREATES THE MAIN MENU TO CONTROL

// ALL THE FUNCTIONS

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void menu::main\_menu()

{ char filename[13]; quiz q; int option; char choice;

clrscr();

do

{

cout<<"\nMAIN MENU";

cout<<"\n1.ADMIN";

cout<<"\n2.PLAY";

cout<<"\n3.QUIT";

cout<<"\nEnter your choice: ";

cin>>option;

switch(option)

{ case 1: password(); admin\_menu(); break;

case 2: do{ strcpy(filename,sub\_menu());

q.play(filename);

} while(strcmp(filename,"null")==0); break;

case 3: exit(0); break;

default: cout<<"\nWrong choice! Please try again!";

getch();

main\_menu();

}

cout<<"\nDo you want to continue(y/n)? ";

cin>>choice;

}while(choice=='y'||choice=='Y');

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// CLASS NAME : menu

// FUNCTION NAME : password

// DETAILS : IT CHECKS WHETHER THE USER HAS

// AUTHORIZED ACCESS OR NOT

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void menu::password()

{ cout<<"\nEnter the password";

char pass[3];

for(inti=0; i<3; i++)

{ pass[i]=getch();

cout<<"\*";

}

pass[i]='\0';

if(strcmp(pass,"fas"))

{ cout<<"\nHurray! You have the authorized access";

return;

}

else

cout<<"\nSorry! You donot have the authorized access";

getch();

exit(0);

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// CLASS NAME : menu

// FUNCTION NAME :sub\_menu

// DETAILS : IT CREATES THE SUB MAIN MENU AND

// RETURN FILE NAME

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

char \*menu::sub\_menu(void)

{ clrscr();

int option; char wrong[13];

cout<<"\nSUB\_MENU";

cout<<"\n1.Computers";

cout<<"\n2.Mathematics";

cout<<"\n3.Science";

cout<<"\n4.General";

cout<<"\n5.Return";

cout<<"\nEnter your choice: ";

cin>>option;

switch(option)

{ case 1: return ("computer.dat"); break;

case 2: return ("maths.dat"); break;

case 3: return ("science.dat"); break;

case 4: return ("general.dat"); break;

case 5: return ("null"); break;

default: cout<<"\nWrong choice! Please try again!";

getch();

strcpy(wrong,sub\_menu());

}

return("null");

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// CLASS NAME : menu

// FUNCTION NAME :admin\_menu

// DETAILS : IT CREATES THE ADMIN MAIN MENU

// FOR ADDING,DELETTING AND MODIFYING

// QUESTIONS

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void menu::admin\_menu()

{ clrscr(); quiz q; char filename[13];

int option;

cout<<"\nADMIN-MENU";

cout<<"\n1.DELETE";

cout<<"\n2.MODIFY";

cout<<"\n3.ADD";

cout<<"\n4.RETURN";

cout<<"\nEnter your choice";

cin>>option;

switch(option)

{ case 1: q.deletion(sub\_menu()) ; break;

case 2: q.modify(sub\_menu()) ; break;

case 3: q.add(sub\_menu()); ; break;

case 4: return;

default: cout<<"\nWrong choice! Please try again!";

getch();

admin\_menu();

}

return;

}

int quiz::retqno()

{ returnques\_no;

}

int quiz::autogen(char filename[13])

{ int r;

fstream file;

file.open(filename,ios::in|ios::out|ios::binary);

if(file.fail())

{ cerr<<"\nThe file cannot be opened";

getch(); exit(1);

}

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

long n=file.tellg();

if(n==0)

r=1;

else

{ file.seekg(-si,ios::end);

file.read((char\*)this,si);

r=retqno();

r++;

}

file.close();

return r;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// CLASS NAME : quiz

// FUNCTION NAME :record\_count

// DETAILS : IT RETURNS THE NO. OF RECORD IN THE

// GIVEN FILE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

int quiz :: record\_count(char filename[13])

{

fstream file ;

if(file.fail())

{ cerr<<"\nThe file cannot be opened";

getch(); exit(1);

}

file.open(filename, ios::in|ios::binary) ;

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

int t=file.tellg();

long n=t/si;

file.close();

return n;

}

void quiz :: add(char filename[13])

{

int valid ;

charch ;

do

{

ques\_no=autogen(filename) ;

clrscr() ;

cout<<"\nquestion # " <<ques\_no ;

cout<<"\nquestion : " ;

do

{

valid = 1 ;

gets(ques) ;

if (strlen(ques) > 159 || strlen(ques) == 0)

{

valid = 0 ;

cout<<"\nLENGTH SHOULD NOT BE BLANK OR GREATER THAN 159" ;

cout<<"\nRetype the question please! "<<endl;

}

} while ( !valid ) ;

cout<<"\nanswer 1 : " ;

do

{

valid = 1;

gets(ans1) ;

if (strlen(ans1) > 49)

{

valid = 0 ;

cout<<"\nLENGTH SHOULD NOT GREATER THAN 49" ;

cout<<"\nRetype the answer please! "<<endl;

}

} while ( !valid ) ;

cout<<"\nanswer 2 : " ;

do

{

valid = 1;

gets(ans2) ;

if (strlen(ans2) > 49)

{

valid = 0 ;

cout<<"\nLENGTH SHOULD NOT GREATER THAN 49" ;

cout<<"\nRetype the answer please! "<<endl;

}

} while ( !valid ) ;

cout<<"\nanswer 3 : " ;

do

{

valid = 1;

gets(ans3) ;

if (strlen(ans3) > 49)

{

valid = 0 ;

cout<<"\nLENGTH SHOULD NOT GREATER THAN 49" ;

cout<<"\nRetype the answer please! "<<endl;

}

} while ( !valid ) ;

cout<<"\nsolution : " ;

do

{

valid = 1 ;

sol = getche() ;

if (sol != '1' && sol != '2' && sol != '3')

{ valid = 0 ;

cout<<"\nPlease enter the solution again! "<<endl;

}

} while ( !valid ) ;

cout<<"\nDo you want to save the record (y/n) : " ;

do

{ valid=1;

ch = getche() ;

ch = toupper(ch) ;

if (ch != 'Y' &&ch != 'N')

{ valid = 0 ;

cout<<"\nPlease enter your choice again! "<<endl;

}

} while ( !valid ) ;

if (ch == 'Y')

{

fstream file ;

file.open(filename, ios::app|ios::binary ) ;

if(file.fail())

{ cerr<<"\nThe file cannot be opened";

getch(); exit(1);

}

file.write((char \*) this, si) ;

file.close() ;

}

cout<<"\nDo you want to add more questions (y/n) : " ;

do

{

valid = 1 ;

ch = getche() ;

ch = toupper(ch) ;

if (ch != 'Y' &&ch != 'N')

{ valid = 0 ;

cout<<"\nPlease enter your choice again!"<<endl;

}

} while ( !valid ) ;

} while (ch == 'Y') ;

}

void quiz :: delete\_record(char filename[13] , intt\_ques\_no)

{ fstream file;

file.open(filename,ios::in|ios::binary);

if(file.fail())

{ cout<<"\nThe file cannot be opened";

getch(); exit(1);

}

fstream temp;

temp.open("temp.dat",ios::out|ios::binary);

if(temp.fail())

{ cout<<"\nThe file cannot be opened";

getch(); exit(1);

}

while(file.read((char\*)this,si))

{ if(ques\_no != t\_ques\_no)

temp.write( (char\*)this , si );

}

file.close(); temp.close();

remove (filename);

rename("temp.dat",filename);

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// CLASS NAME : quiz

// FUNCTION NAME :display\_record

// DETAILS : IT DISPLAY THE RECORD FOR THE GIVEN

// QUESTION NO. FROM THE GIVEN FILE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void quiz :: display\_record(char filename[13], intt\_ques\_no)

{

fstream file ;

file.open(filename, ios::in|ios::binary) ;

if(file.fail())

{ cerr<<"\nThe file cannot be opened";

getch(); exit(1);

}

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

while (file.read((char \*) this, si))

{

if (t\_ques\_no == ques\_no)

{

cout<<"\nquestion # " <<t\_ques\_no ;

cout<<"\nquestion : " <<ques ;

cout<<"\nanswer 1 : " <<ans1 ;

cout<<"\nanswer 2 : " <<ans2 ;

cout<<"\nanswer 3 : " <<ans3 ;

break ;

}

}

file.close() ;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// CLASS NAME : quiz

// FUNCTION NAME :found\_record

// DETAILS : IT SEARCH THE GIVEN RECORDS IN THE

// GIVEN FILE AND RETURN 0 IF NOT FOUND

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

int quiz :: found\_record(char filename[13], intt\_ques\_no)

{

fstream file ;

file.open(filename, ios::in|ios::binary) ;

if(file.fail())

{ cerr<<"\nThe file cannot be opened";

getch(); exit(1);

}

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

int found=0 ;

while (file.read((char \*) this, si))

{

if (t\_ques\_no == ques\_no)

{

found++ ;

break ;

}

}

file.close() ;

return found ;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// CLASS NAME : quiz

// FUNCTION NAME : deletion

// DETAILS : IT GIVE THE QUESTION NO. TO DELETE

// RECORD FROM THE GIVEN FILE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void quiz :: deletion(char filename[13])

{ intt\_ques\_no=0; char choice;

cout<<"\nEnter the question number to be deleted: ";

cin>>t\_ques\_no;

if (t\_ques\_no == '0')

return ;

if (!found\_record(filename,t\_ques\_no))

{

cout<<"\nQuestion not found" ;

cout<<"\nPress any key to continue..." ;

getch() ;

return ;

}

clrscr();

display\_record(filename,t\_ques\_no) ;

cout<<"\nDo you want to delete this record(y/n)";

cin>>choice;

choice=toupper(choice);

if(choice=='Y')

{ cout<<"\nWait.....";

delete\_record(filename,t\_ques\_no);

cout<<"\nThis record has been deleted";

cout<<"\nPress any key to continue...";

getch();

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// CLASS NAME : quiz

// FUNCTION NAME :modify\_record

// DETAILS : IT MODIFY THE RECORD FOR THE GIVEN

// QUESTION NO. FROM THE GIVEN FILE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void quiz :: modify\_record(char filename[13], intt\_ques\_no)

{

int valid, modified=0 ;

charch ;

cout<<"\nQUESTION # " <<t\_ques\_no ;

cout<<"\nQUESTION : " ;

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

do

{

valid = 1 ;

ch = getche() ;

ch = toupper(ch) ;

if (ch != 'Y' &&ch != 'N')

{

valid = 0 ;

}

} while (!valid) ;

valid = 0 ;

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

{

modified = 1 ;

valid = 1 ;

gets(ques) ;

if (strlen(ques) >159 || strlen(ques) == 0)

{

valid = 0 ;

cout<<"\nLENGTH SHOULD NOT BLANK OR GREATER THAN 159" ;

cout<<"\nPress any key to continue..." ;

getch() ;

}

}

cout<<"\nANSWER 1 : " ;

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

do

{

valid = 1 ;

ch = getche() ;

ch = toupper(ch) ;

if (ch != 'Y' &&ch != 'N')

{

valid = 0 ;

}

} while (!valid) ;

valid = 0 ;

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

{

modified = 1 ;

valid = 1 ;

gets(ans1) ;

if (strlen(ans1) > 49)

{

valid = 0 ;

cout<<"\nLENGTH SHOULD NOT GREATER THAN 49" ;

cout<<"\nPress any key to continue..." ;

getch() ;

}

}

cout<<"\nANSWER 2 : " ;

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

do

{

valid = 1 ;

ch = getche() ;

ch = toupper(ch) ;

if (ch != 'Y' &&ch != 'N')

{

valid = 0 ;

}

} while (!valid) ;

valid = 0 ;

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

{

modified = 1 ;

valid = 1 ;

gets(ans2) ;

if (strlen(ans2) > 49)

{

valid = 0 ;

cout<<"\nLENGTH SHOULD NOT GREATER THAN 49" ;

cout<<"\nPress any key to continue..." ;

getch() ;

}

}

cout<<"\nANSWER 3 : " ;

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

do

{

valid = 1 ;

ch = getche() ;

ch = toupper(ch) ;

if (ch != 'Y' &&ch != 'N')

{

valid = 0 ;

}

} while (!valid) ;

valid = 0 ;

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

{

modified = 1 ;

valid = 1 ;

gets(ans3) ;

if (strlen(ans3) > 49)

{

valid = 0 ;

cout<<"\nLENGTH SHOULD NOT GREATER THAN 49" ;

cout<<"\nPress any key to continue..." ;

getch() ;

}

}

if (!modified)

return ;

cout<<"\nSOLUTION : " ;

do

{

valid = 1 ;

sol = getche() ;

if (sol != '1' && sol != '2' && sol != '3')

{

valid = 0 ;

}

} while (!valid) ;

cout<<"\nDo you want to save the record (y/n) : " ;

do

{

valid = 1 ;

ch = getche() ;

ch = toupper(ch) ;

if (ch != 'Y' &&ch != 'N')

valid = 0 ;

} while ( !valid ) ;

if (ch == 'N')

return ;

fstream file ;

file.open(filename, ios::out | ios::ate|ios::binary) ;

if(file.fail())

{ cerr<<"\nThe file cannot be opened";

getch(); exit(1);

}

intrecno ;

recno = t\_ques\_no ;

int location ;

location = (recno-1) \* si ;

file.seekp(location) ;

file.write((char \*) this, si) ;

file.close() ;

clrscr() ;

cout<<"\nRecord Modified" ;

cout<<"\nPress any key to continue..." ;

getch() ;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// CLASS NAME : quiz

// FUNCTION NAME : modify

// DETAILS : IT GIVE THE QUESTION NO. TO MODIFY

// RECORD FROM THE GIVEN FILE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void quiz :: modify(char filename[13])

{

clrscr() ;

int valid ;

intt\_ques\_no=0, t;

charch ;

cout<<"\nEnter question no. to be modify : " ;

do

{

cin>>t\_ques\_no ;

} while (t\_ques\_no==0);

if (!found\_record(filename,t\_ques\_no))

{

cout<<"\nquestion not found" ;

cout<<"\nPress any key to continue..." ;

getch() ;

return ;

}

clrscr() ;

display\_record(filename,t\_ques\_no) ;

cout<<"\nDo you want to modify this Record (y/n) : " ;

do

{

valid = 1 ;

ch = getche() ;

ch = toupper(ch) ;

if (ch != 'Y' &&ch != 'N')

valid = 0 ;

} while (!valid ) ;

if (ch == 'N')

return ;

modify\_record(filename,t\_ques\_no) ;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// CLASS NAME : quiz

// FUNCTION NAME: display\_score

// DETAILS : IT DISPLAY THE SCORE FOR THE GIVEN

// PLAYER'S NAME

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void quiz :: display\_score(char name[20], int played, int correct)

{

clrscr() ;

cout<<"\nS C O R E B O A R D";

cout<<"\nPlayer's Name : " <<name ;

cout<<"\nquestions Attempted : " <<played ;

cout<<"\nCorrect answer : " <<correct ;

cout<<"\nWrong answer : " <<played-correct ;

int score ;

score = correct \* 10 ;

cout<<"\nScore : " <<score ;

inttplayed ;

tplayed = played \* 10 ;

float percentage ;

percentage = ((score\*100)/tplayed) ;

cout<<"\nPercentage : " <<percentage ;

getch() ;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// CLASS NAME : quiz

// FUNCTION NAME : play

// DETAILS : IT CREATES SCREEN FOR PLAYING QUIZ

// TO THE PLAYER FOR THE GIVEN FILE NAME

// AND KEEP SCORE.

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void quiz :: play(char filename[13])

{

clrscr() ;

int valid ;

char name[20] ;

cout<<"\nEnter your name : " ;

do

{

valid = 1 ;

gets(name) ;

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

return ;

if (strlen(name) > 18)

{

valid = 0 ;

cout<<"\nLENGTH SHOULD NOT GREATER THAN 18" ;

cout<<"\nPress any key to continue..." ;

getch() ;

}

} while ( !valid ) ;

chart\_sol, ch ;

int correct=0, played=0 ;

fstream file ;

file.open(filename,ios::in|ios::binary) ;

if(file.fail())

{ cerr<<"\nThe file cannot be opened";

getch(); exit(1);

}

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

while (file.read((char \*) this, si))

{

clrscr() ;

cout<<"\nSCORE:" <<correct\*10 ;

valid = 1 ;

cout<<"\nquestion # " <<ques\_no ;

cout<<"\nquestion : " <<ques ;

cout<<"\nanswer 1 : " <<ans1 ;

cout<<"\nanswer 2 : " <<ans2 ;

cout<<"\nanswer 3 : " <<ans3 ;

cout<<"\nEnter correct solution : " ;

do

{ valid=1;

t\_sol = getche() ;

if (t\_sol == 27)

{

if (played)

display\_score(name,played,correct) ;

return ;

}

if (t\_sol != '1' &&t\_sol != '2' &&t\_sol != '3')

valid = 0 ;

} while (!valid) ;

played++ ;

if (t\_sol != sol)

{

cout<<"\nwrong answer" ;

cout<<"\nCorrect answer : " <<sol ;

}

else

{

correct++ ;

cout<<"\nGOOD ! CORRECT ANSWER" ;

cout<<"\nSCORE:" <<correct\*10 ;

ch = getch() ;

}

if (ch == 27)

{

display\_score(name,played,correct) ;

return ;

}

}

cout<<"\nNo more questions available" ;

getch() ;

if (!played)

return ;

display\_score(name,played,correct) ;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION NAME : main

// DETAILS : IT CALLS THE START FUNCTION AND

// MENU FUNCTION.

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void main()

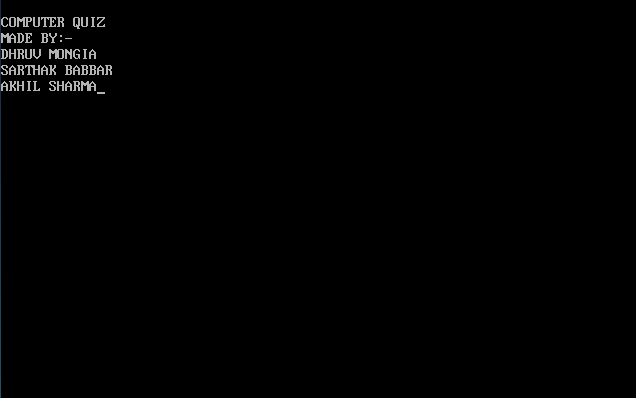
{ menu m;

m.start();

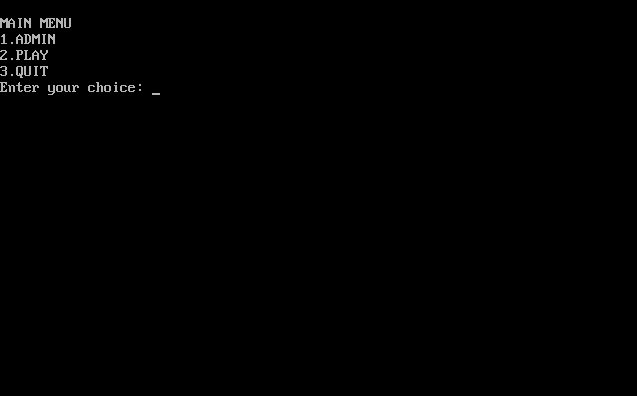
}

Outputs

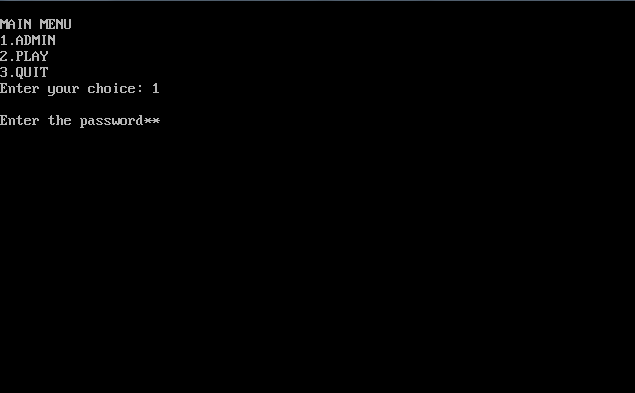
1.Start



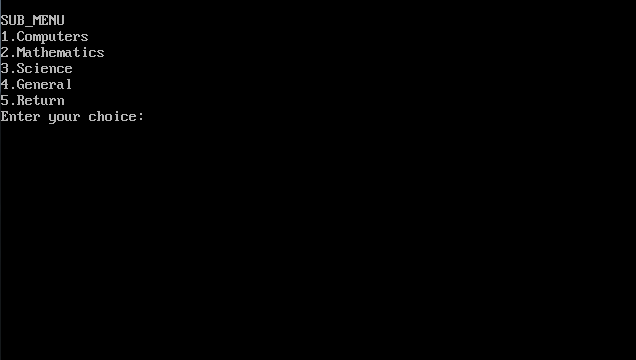
2.Main Menu--ADMIN

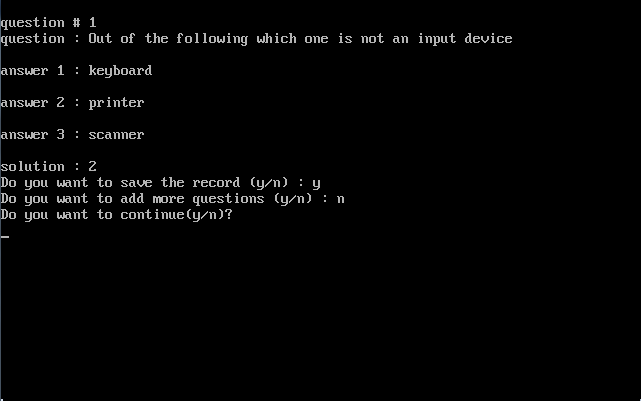


3.ADMIN-PASS

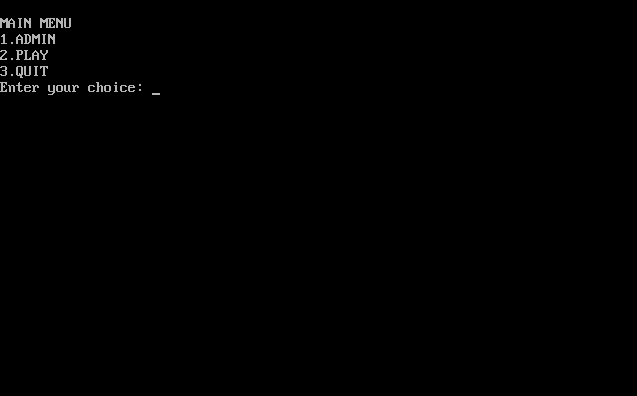


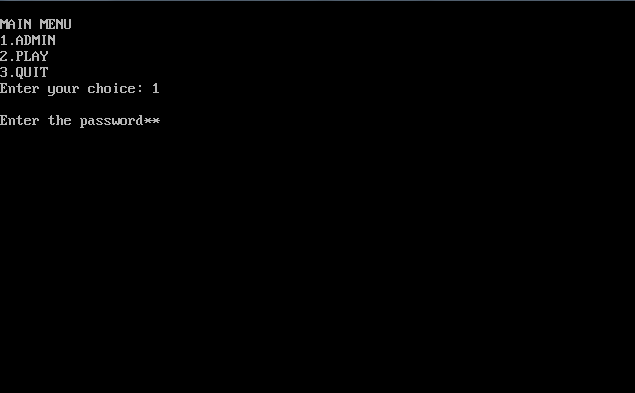
4.ADMIN--ADD

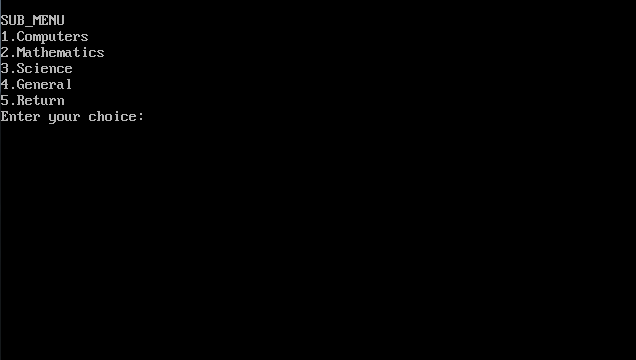


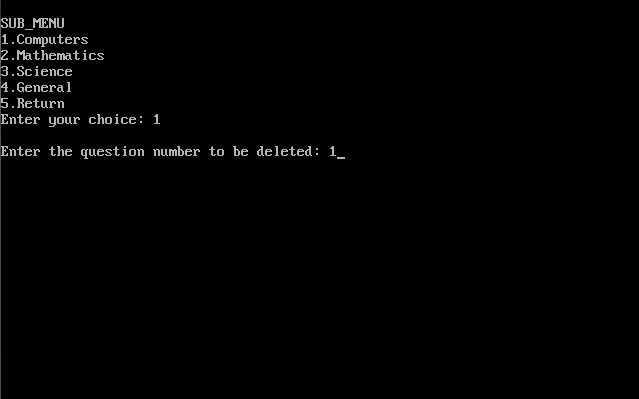


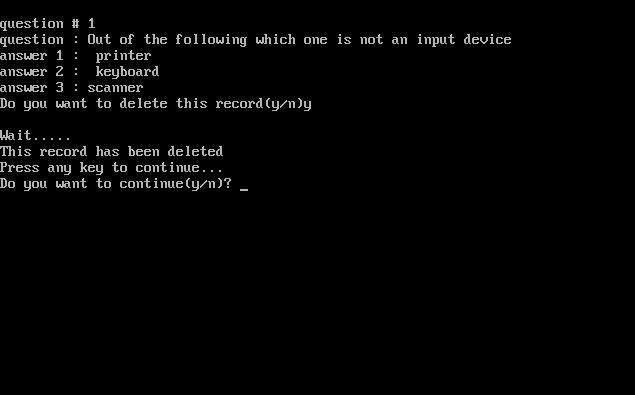
5.DELETE



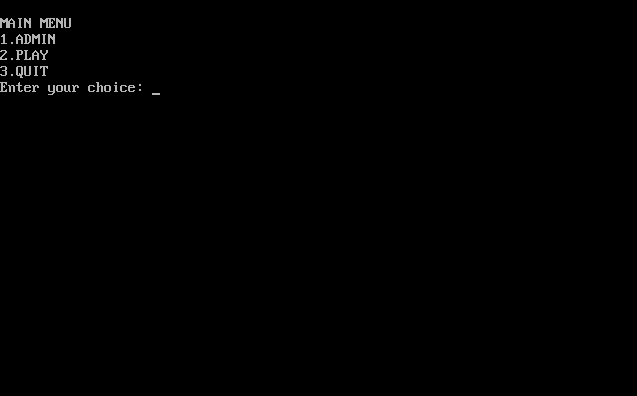


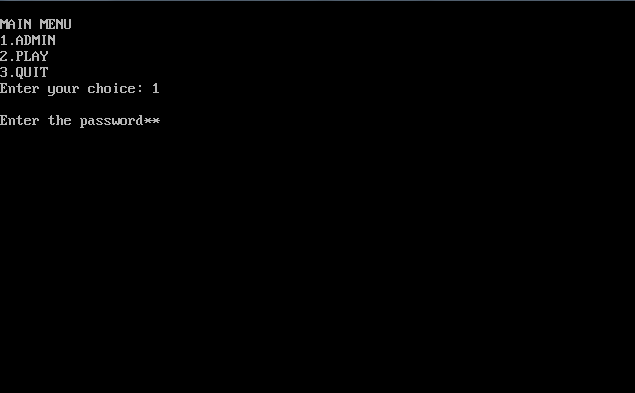


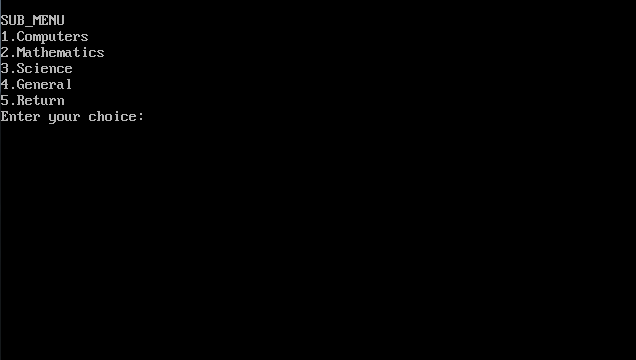


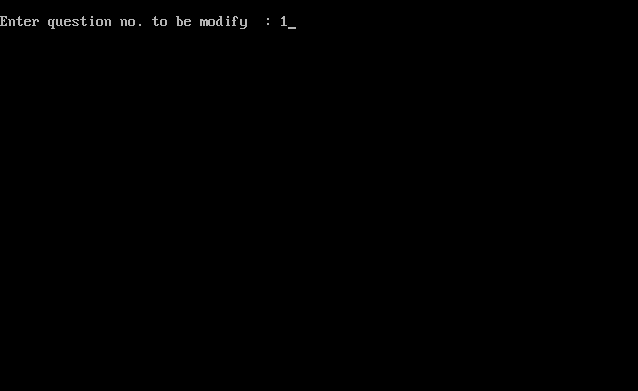


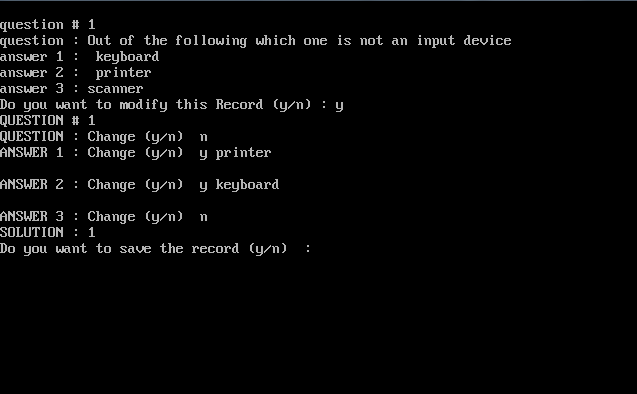
6.MODIFY





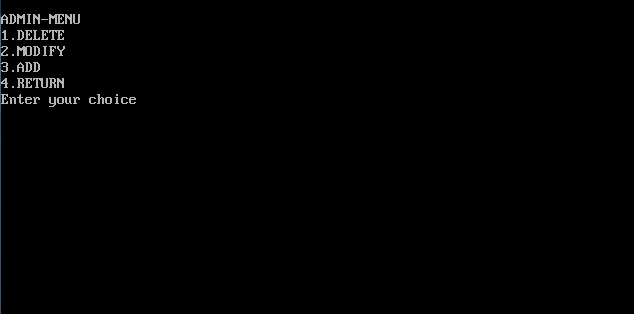


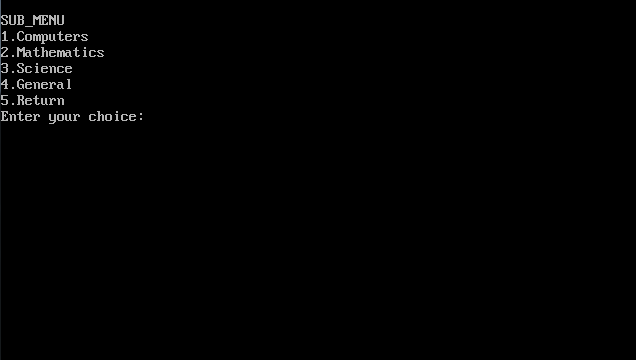


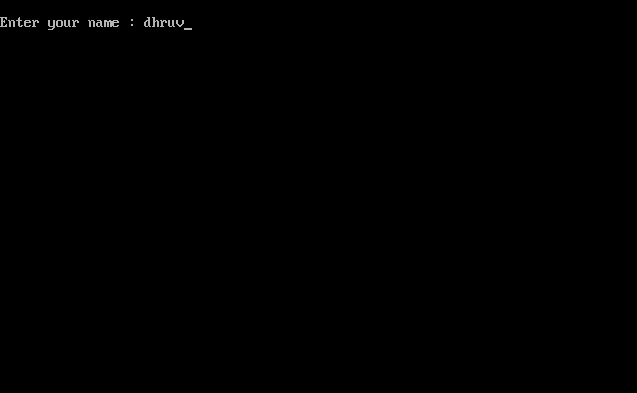


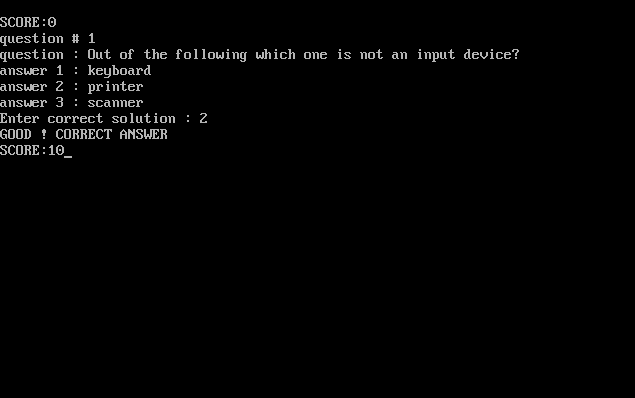


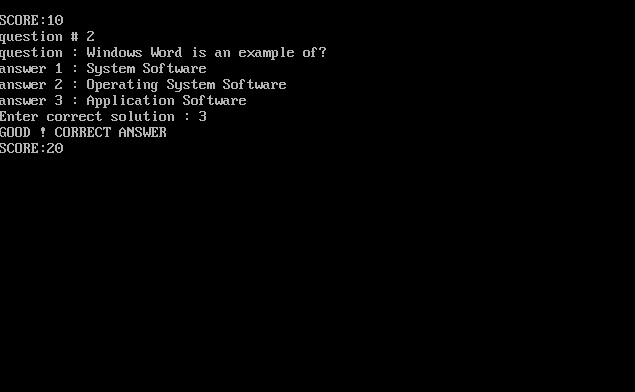
7.PLAY

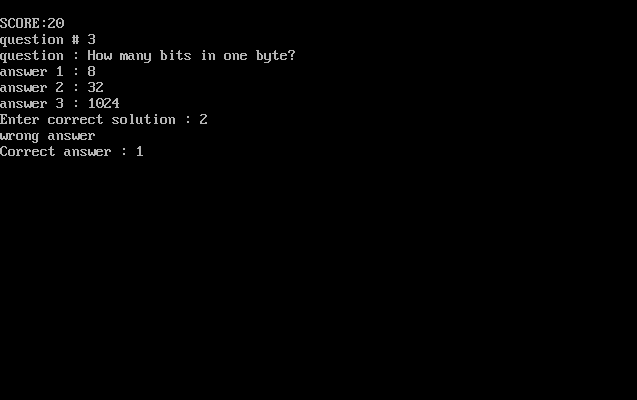


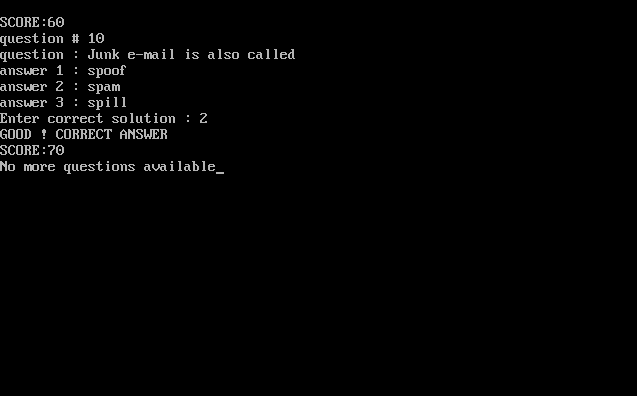


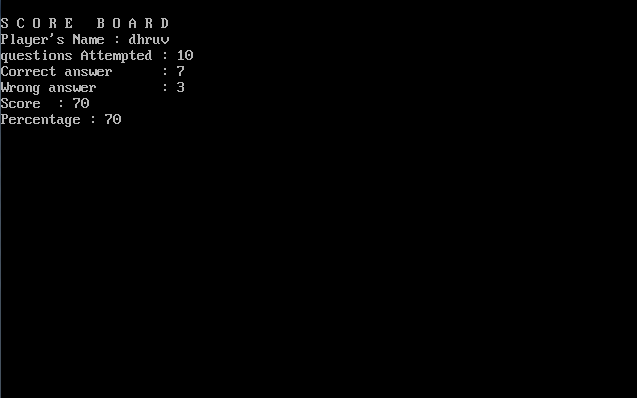












BIBLIOGRAPHY

* www.google.co.in
* www.yahoo.com
* Together With(Class 12th Book )
* http://www.classmarker.com