Introduction to programming using c

Project Report on

“QUIZ GAME”

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

ABSTRACT

Quiz game is a project designed to allow users to play quiz with various types of questions. This project helps to avoid manual problems and erases the work of conducting a quiz game. This project uses the C programming language and contains different types of questions with answers in the form of multiple choices. The user must enter the name before they start the quiz game. The answers will be displayed on the screen, after the completion of all questions. The program quiz game is built to help players have a great experience rather than playing it manually. The player can exist from the game, after the score is displayed or the player can also play the game again.

CONTENTS

Chapter No.	Chapter Name	Page
1	Introduction	1
2	Objective	2
3	System Requirement specifications	3
3.1	Hardware specifications	
3.2	Software specifications	
4	System Design	4
5	System Implementation	7
6	Results	10

INTRODUCTION

Quiz game is one of the best ways to test players' knowledge and it’s also an escape from their busy lives. Most of the people these days are very much interested in playing online games. The quiz game program is a mini C programming project, which provides easy-to-attempt and comprehends instructions regarding the game. In this project, a number of questions are asked, and the user is awarded a cash prize for each correct answer given. The Quiz game program helps collect accurate answers and it is easy to operate. After entering the name, the player is redirected to a menu which typically consists of the rules of the game. The tips are also provided for the better understanding of the game. The details about the prize are also mentioned. It consists of various questions by which to better understand the game. The details about the prize are also mentioned. It consists of various questions by which the user can win prize money by answering them accurately. Once the questions are completed, based on the answers given by the player, results are displayed.

OBJECTIVE

The main objective of Quiz Game is to facilitate a user friendly environment for all users and to reduce the manual effort and also to enhance the user's experience and talents. The project enables switch case allowing multiple questions and also displays the result on the console. The program utilizes most of the important concepts in C. In past days Quiz is conducted manually but in further resolution of the technology we are able to generate the score and pose the queries automatically. The functional requirements include to create users that are going to participate in the quiz, automatic score and report generation and administrative tasks like add, delete, update for admin privilege users. In this application, all the permissions lies with the administrator i.e., specifying the details of quiz, displaying questions and answers accordingly, displaying score for the game played and generate the overall report.

SYSTEM REQUIREMENT SPECIFICATIONS

*** SOFTWARE REQUIREMENTS: -**

Language used: C

Operating System: Windows 7,8,10,11

Compiler: IDE: Code blocks

***HARDWARE REQUIREMENTS: -**

Hard disk: 1Tb

SSD: 512 Gb

Processor: Intel i3(minimum)

SYSTEM DESIGN

Algorithm:

STEP-1: Start

STEP-2: Program display menu to the user to start, help or to exit.

STEP-3: According to the user's input, steps are implemented.

STEP-4: If the user input is 1, it asks the user's credentials.

STEP-5: If the user input is 2, it displays the help void.

STEP-6: If the user input is 3, the program will quit.

STEP-7: The questions are displayed one after another, as soon as the user inputs 1.

STEP-8: If answered correctly, points are awarded i.e., the score is increased.

STEP-9: If answered wrong, points are not awarded i.e., the score remains same.

STEP-10: If the entered answer is wrong, it displays correct answer.

STEP-11: Step-8 and step-9 are repeated till the questions are completed.

STEP-12: After the completion of the questions, the number of correct answers are counted.

STEP-13: Displays the score, rating for the game played.

STEP-14: It asks the user whether to play the game or quit.

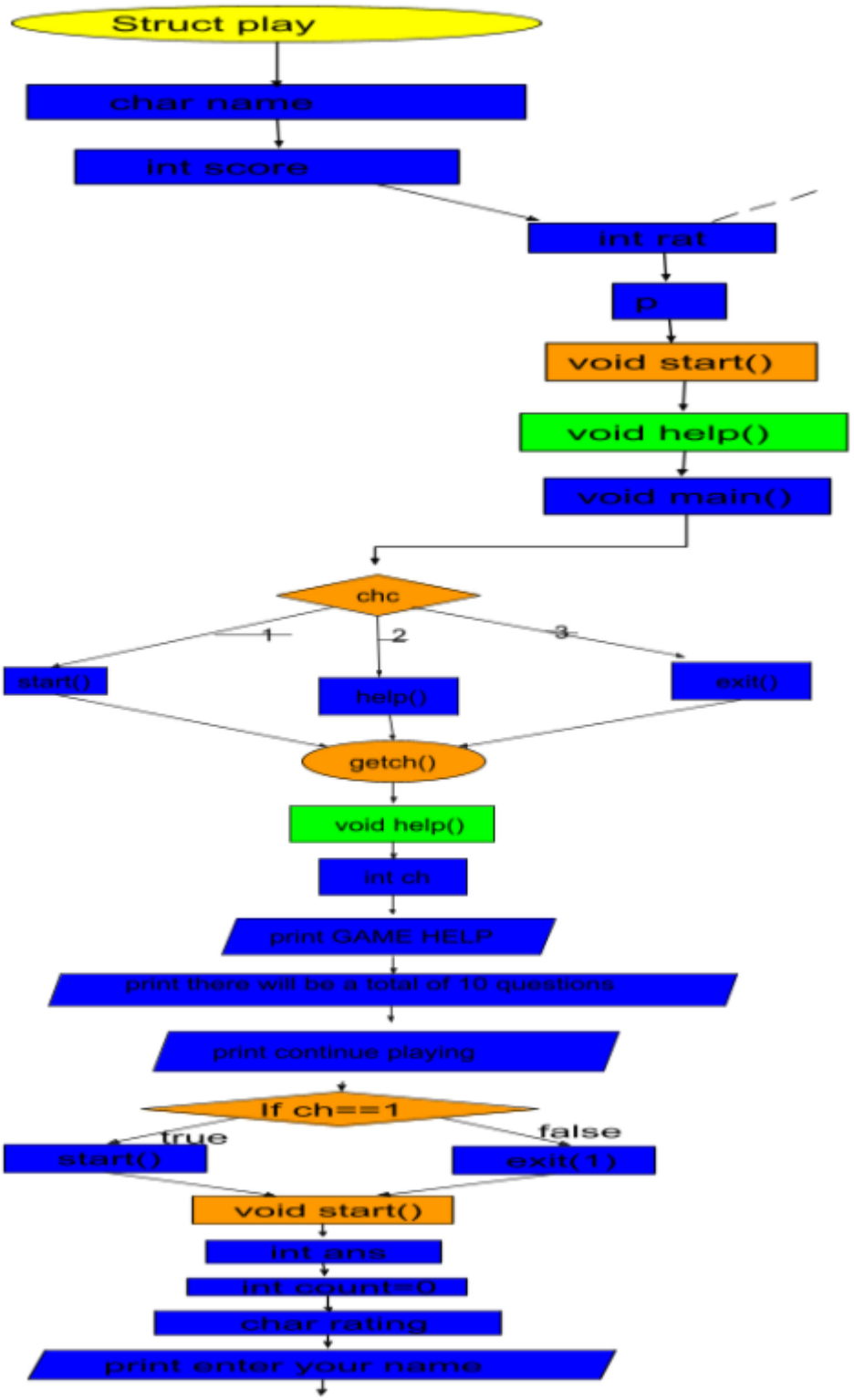
STEP-15: If the user chooses to play again, step-8 to step-15 are repeated.

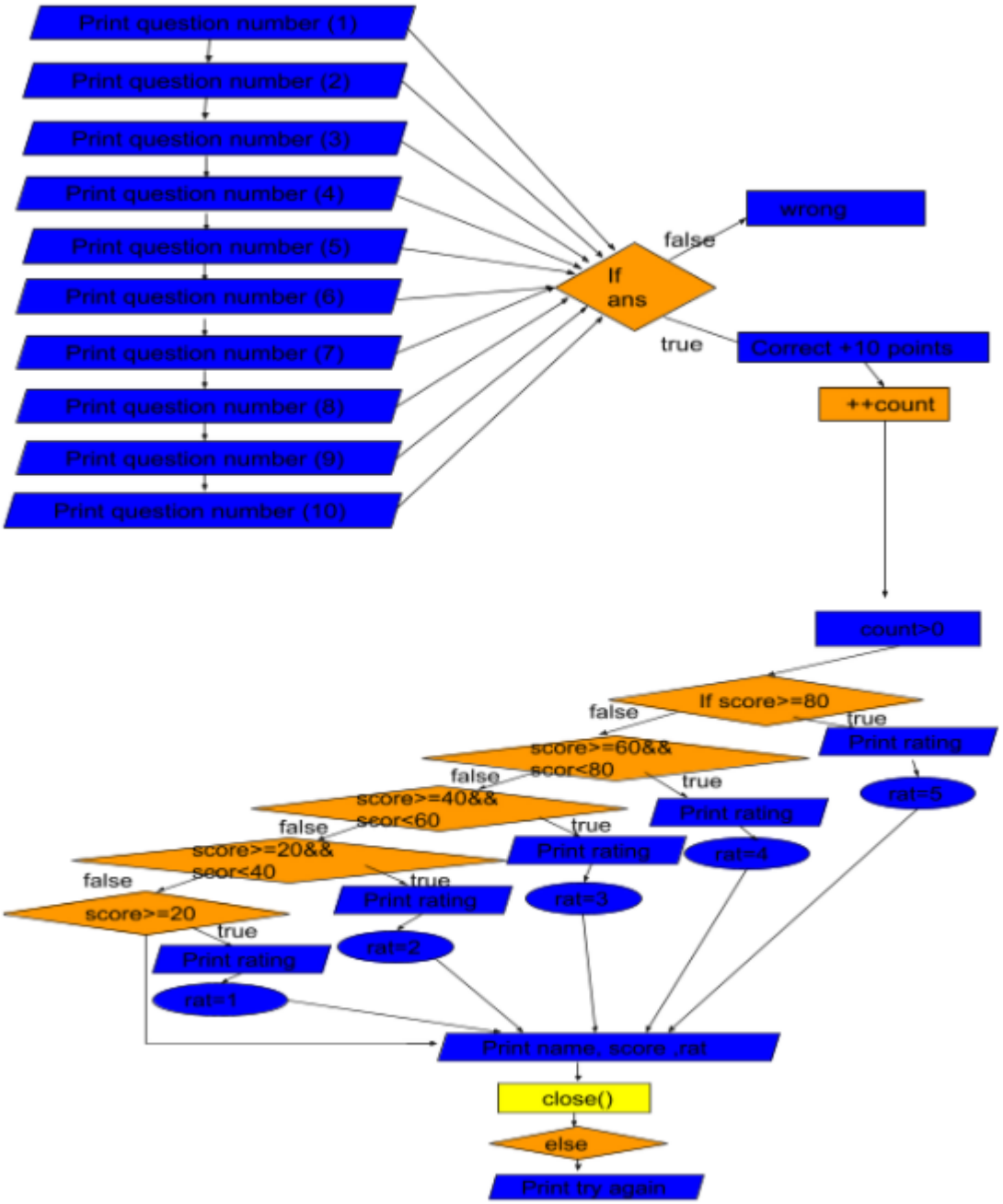
STEP-16: If user chooses to quit the game, step-6 takes place.

STEP-17: End of the program

STEP-18: Stop

Flowchart:


Struct play
char name
int score
int rat
p
void start()
void help()
void main()
chc
1
2
3
start()
help()
exit()
getch()
void help()
int ch
print GAME HELP
print there will be a total of 10 questions
print continue playing
If ch==1
true
false
start()
exit(1)
void start()
int ans
int count=0
char rating
print enter your name


Print question number (1)
Print question number (2)
Print question number (3)
Print question number (4)
Print question number (5)
Print question number (6)
Print question number (7)
Print question number (8)
Print question number (9)
Print question number (10)
If ans
false
wrong
true
Correct +10 points
++count
count>0
If score>=80
false
true
Print rating
rat=5
score>=60&& scor<80
false
true
Print rating
rat=4
score>=40&& seor<60
false
true
Print rating
rat=3
score>=20&& seor<40
false
true
Print rating
rat=2
score>=20
false
true
Print rating
rat=1
Print name, score ,rat
close()
else
Print try again

SYSTEM IMPLEMENTATION

The project is implemented using simple c programming classes implemented as follows:

•HEADER FILES:

***stdio.h** - This is a standard input/output header file. It contains all the library functions regarding standard input/output.

***stdlib.h** - This header file contains all the general library functions like malloc (), calloc (), exit (), etc.

***conio.h** - This is a console input/output header file.

•STRUCTURE:

A structure is a user defined data type in C which creates a data type that can be used to group items of possibly different types into a single type. 'struct' keyword is used to create a structure.

Usage: we used "struct play "in this program.

Syntax:

```
struct structure_name

{

   data_type member1;

   data_type member2;

   .

   .

   data_type memeberN;

}
```

•SWITCH CASE:

The switch statement in C is an alternate to if-else-if ladder statement which allows us to execute multiple operations for the different possible values of a single variable called switch variable.

Usage: We are using switch case as it allows a variable to be tested for equality against a list of values.

Syntax:

```
Switch (expression)

{

case value1:    //code to be executed;

 break; //optional

case value2:       //code to be executed;

break; //optional

.....

default:

code to be executed if all cases are not matched;

}
```

•FUNCTIONS:

A function is a group of statements that together perform a task.

Usage: We use functions in a program to avoid rewriting same logic/code again and again in a program.

•VOID:

A void is a keyword that references an empty data type that has no return value.

Usage: We use the void data type in a program when we don't want to return any value to the calling function.

The three void datatypes we used in this project are:

1.) Void main ()

2.) Void start ()

3.) Void help ()

RESULTS

Fig-2.1 User input

Fig-2.2 Game Help (if we press 2)

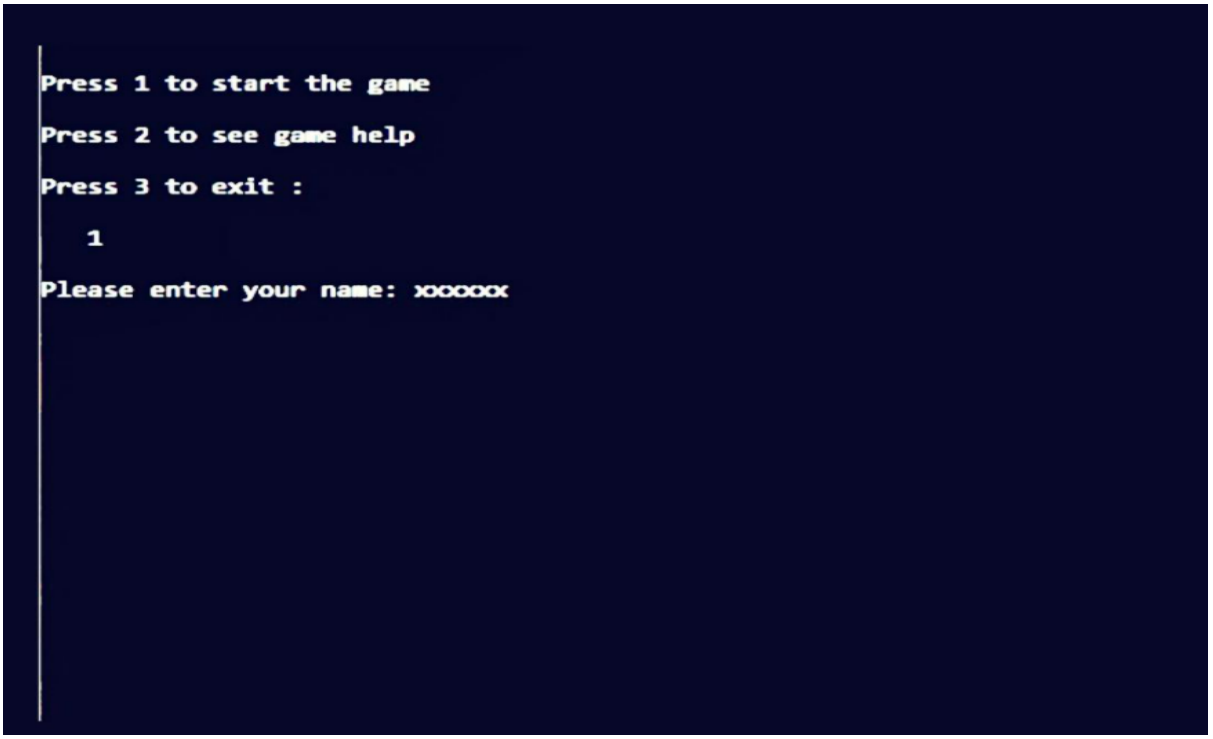


Fig-2.3 Enter your name

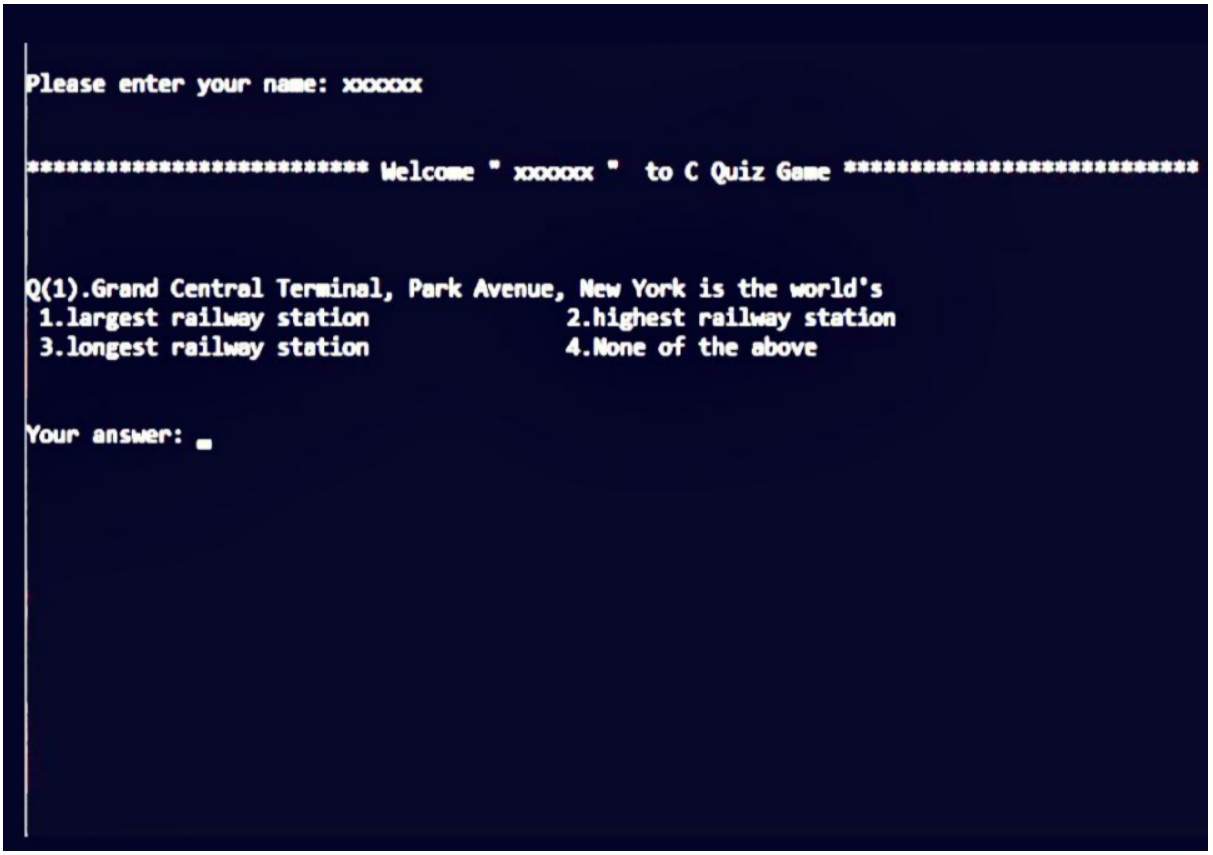


Fig-2.4 Choose any 1 option among the 4

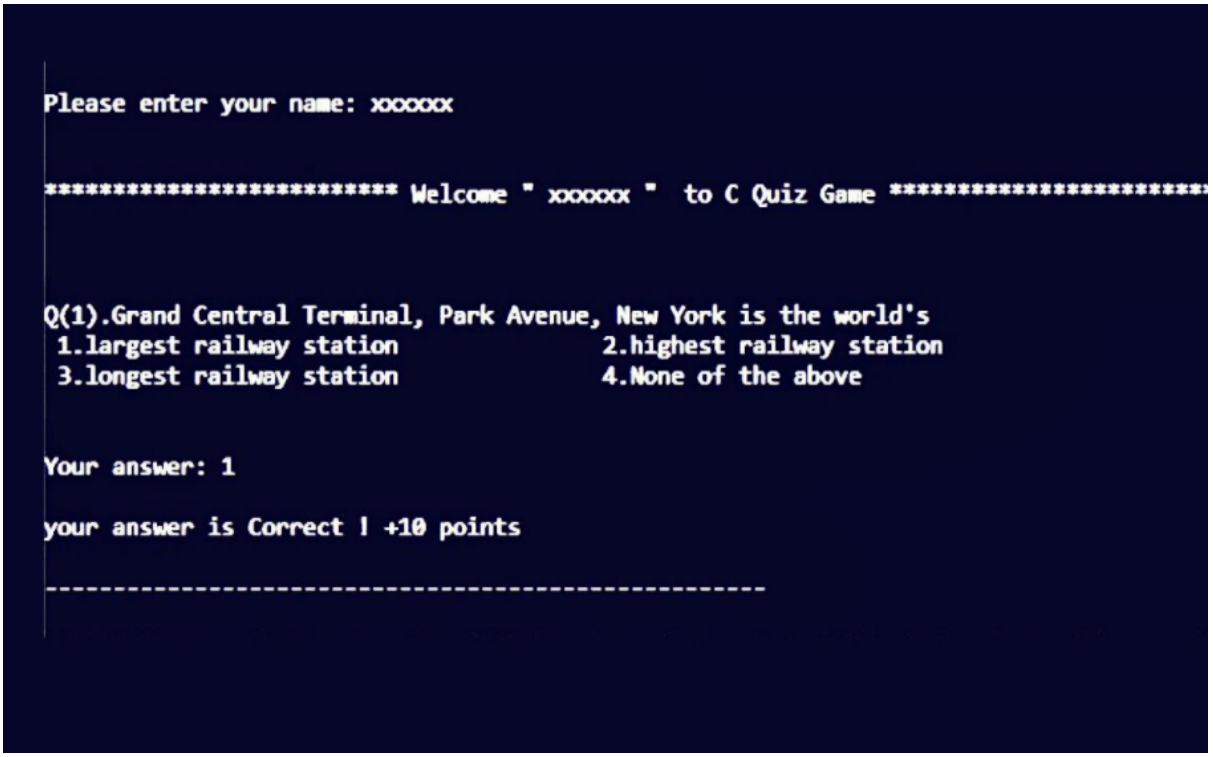


Fig-2.5 If the answer is correct

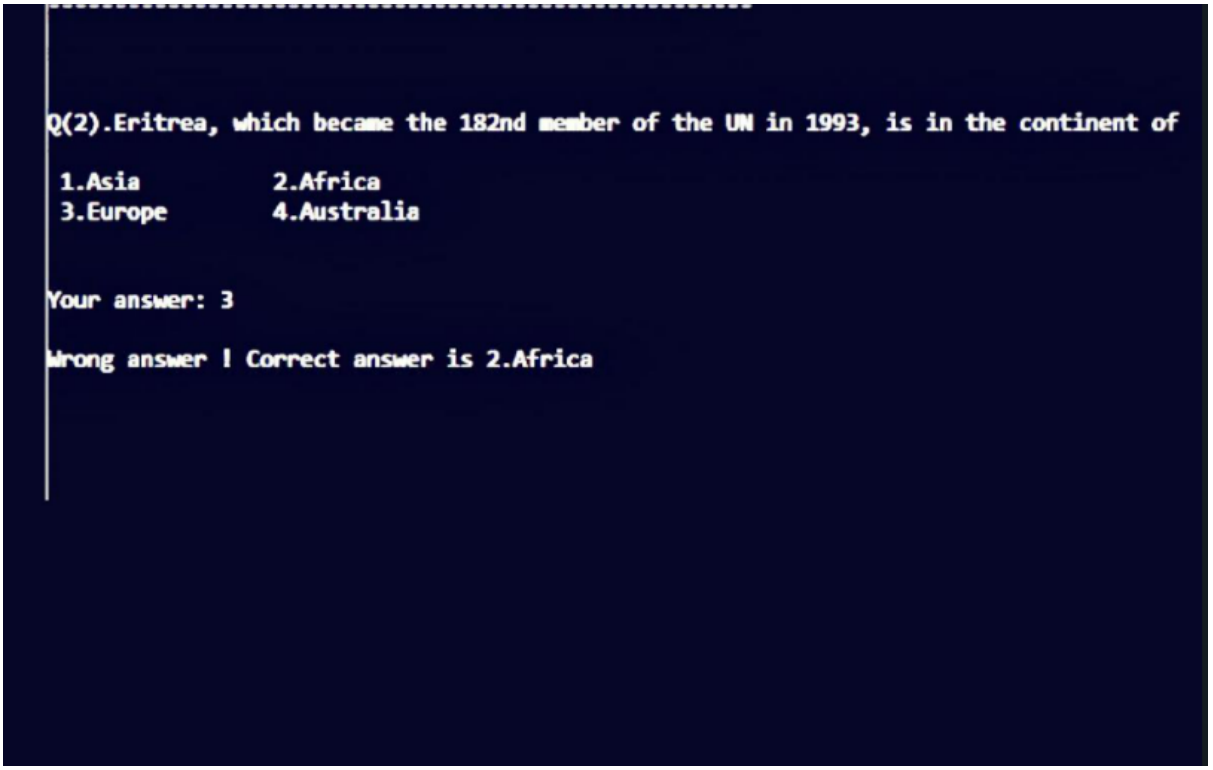


Fig-2.6 If the answer is wrong

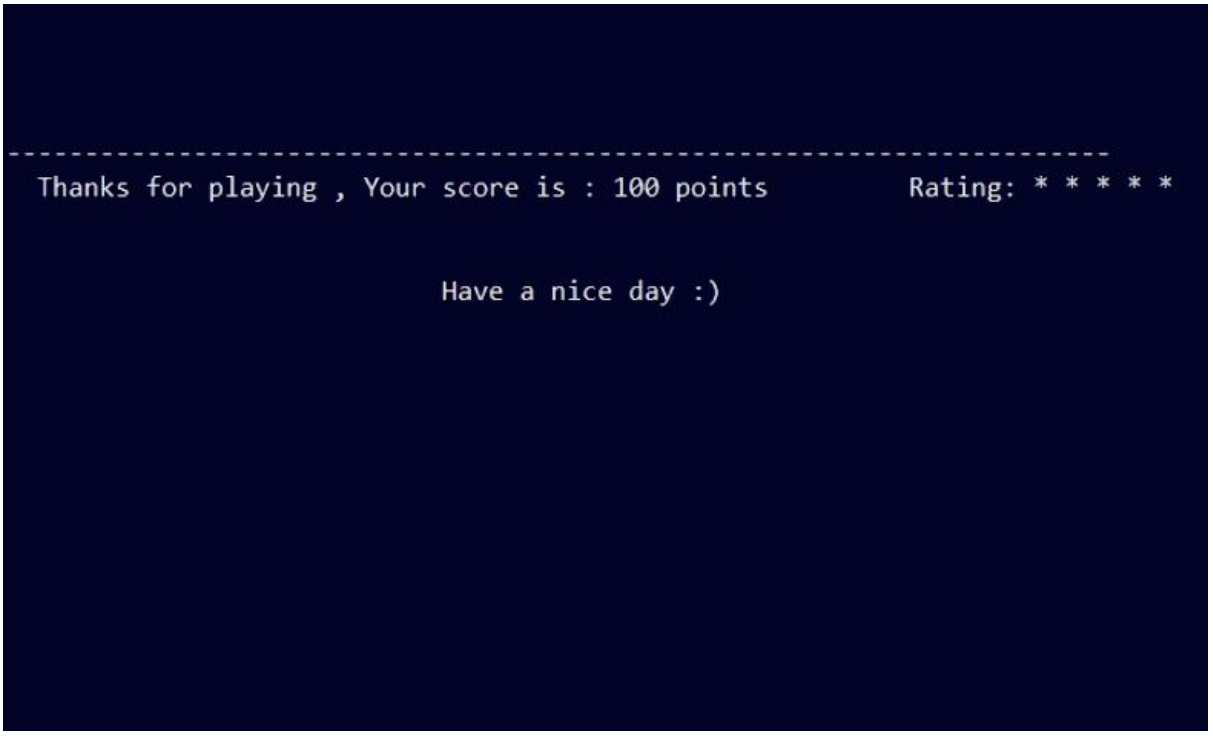


Fig-2.7 After the completion of all questions