-

KAUN BANEGA CROREPATI GAME

NAME -ARYAN

UPADHYAY

SAP ID -590027715

Abstract

This project presents a console-based implementation of the popular quiz show **Kaun Banega Crorepati (KBC)** using the C programming language. The system features 12 multiple-choice questions, increasing prize money, two lifelines (50–50 and Skip), timer functionality, modularized code structure using multiple files, and clean output formatting.
The project demonstrates key programming concepts including modular programming, function separation, file structuring, arrays, conditional logic, and user interaction. The system is designed to be simple, efficient, and easy to extend with new features like additional lifelines or a GUI.

Contents

1 Problem Definition

1.1 Overview

The **KBC Game Project** simulates the real TV show experience through a console interface. The player answers a series of MCQs, uses lifelines strategically, and attempts to win the final amount.
This project is ideal for learning modular programming in C, user input handling, control flow, and structuring real-world logic into programs.

1.2 Objectives

The major objectives of this project are:

- To design a quiz game inspired by KBC using C programming.
- To implement modular program structure using **header files**, **multiple C files**, and **function definitions**.
- To provide an interactive gameplay mechanism with lifelines.

- To demonstrate real-time programming using the **time.h** library.
- To improve user experience with well-structured options and output.
- To develop a system that can be easily expanded or integrated into a graphical version.

2 System Design

2.1 System Architecture

The architecture follows a **Modular Layered Structure**:

- **Presentation Layer**:

Handles all interaction with the user:

- Displaying questions & options
- Showing prize money
- Asking for lifeline usage
- Displaying results & summary
- Printing messages, separators, and flow

- **Business Logic**:

contains the game logic:

- Checking if answers are correct
- Applying lifeline effects
- Timer calculation
- Moving to next question or ending game
- Maintaining score and progress
- Handling wrong answer logic
- Prize money evaluation

- **Data Access**:

Stores and supplies data:

- Questions
- Options
- Correct answers array
- Lifeline removal option sets (50–50)
- Prize money array

2.2 Data Structures:

1 Arrays

Questions Array

char *questions[12];

Stores all 12 questions.

Options Array

char *options[12][4];

Stores 4 options for each question.

Correct Answers

int correctOption[12];

Stores correct option number for each question.

50-50 Lifeline Arrays

int fifty1[12], fifty2[12];

Defines which two options remain after 50-50.

Prize Money Array

long prizeMoney[13];

2 Implementation Details

2.1 Key Features

 12 MCQ questions

 ✓ Two lifelines

- 50-50
- Skip

 ✓ Timer for each question

 ✓ Prize money structure

 ✓ Game ends on wrong answer

 ✓ Final summary

 ✓ Clean output formatting

 ✓ Multi-file modular architecture

 ✓ Easy to extend

3 Testing & Results

1 Test Case 1 – Correct Input

Input: Correct answers sequentially
Expected Result: Player progresses until Q12
Result: ✓ Passed

2 Test Case 2 – Wrong Answer

Input: Wrong answer at Q3
Expected: Game over message
Result: ✓ Passed

3 Test Case 3 – 50-50 Lifeline

Input: Use 50-50 at Q2
Expected: 2 options displayed
Result: ✓ Passed

4 Test Case 4 – Skip Lifeline

Input: Skip Q4
Expected: Moves to Q5 without penalty
Result: ✓ Passed

5 Test Case 5 – Timer Accuracy

Action: Delay answering intentionally
Expected: Time difference shows correctly
Result: ✓ Passed

Conclusion & Future Work

5.1 Conclusion

The KBC Game Project successfully demonstrates modular programming and interactive console design using C.
The game is functional, stable, and efficiently handles lifelines, timers, and logic flows. The project showcases understanding of arrays, conditional logic, multi-file management, and real-time functions.

5.2 Future Work

- Add graphical UI using Java / Python.
- Add more lifelines like Audience Poll, Flip the Question.
- Add sound effects like real KBC.
- Save scores to external files.
- Add database support for large question banks.
- Add online multiplayer mode.

6 References

1. **K&R C Programming Language**, Brian Kernighan & Dennis Ritchie
2. TutorialsPoint – C Programming Reference
3. GeeksForGeeks – Flowcharts & Program Structures
4. Online C documentation and compiler references

Compilation and Execution c main.c functions.c -o kbc_game