CHANDIGARH UNIVERSITY
Discover. Learn. Empower.

UNIVERSITY INSTITUTE OF COMPUTING

PROJECT REPORT
ON
Project name

Program Name: BCA

Subject Name/Code:

Computing Aptitude/23CAP-308

Submitted by:

Name: Kajal kumari

UID: 23BCA10770

Section: 23BCA-2 B

Submitted to:

Name: Mr. Suraj Prakash

Designation: Assistant Prof.

1. Title

Quiz Application using C++ Programming Language

This project is a simple console-based program that allows users to answer multiplechoice questions and receive instant feedback along with their total score at the end.

---

2. Aim of the Project

The main aim of this project is to develop a **Quiz Application** using **C++**, which evaluates the user's knowledge on different topics. The system asks a series of questions, takes user input, checks the correctness of each answer, and displays the total marks obtained.

---

3. Objectives

- To create an interactive quiz game using basic C++ concepts.
- To enhance user engagement by providing immediate feedback.
- To demonstrate the use of arrays, loops, conditional statements, and functions in a real-world scenario.
- To strengthen logical thinking and programming fundamentals.
- To design a lightweight, text-based quiz that runs on any C++ compiler.

---

4. Tools and Technologies Used

Component	Description
Programming Language	C++
Compiler	GCC / Turbo C++
IDE (Optional)	Code::Blocks / Dev C++ / Visual Studio Code
Component	**Description**
Operating System	Windows 10 or higher

Concepts Used Arrays, Loops, Conditional Statements, Input/Output Handling

---

5. System Requirements

Hardware Requirements:

- Processor: Intel Core i3 or higher
- RAM: Minimum 2 GB
- Hard Disk: 50 MB free space
- Display: Standard VGA **Software Requirements:**
- Operating System: Windows 10 or higher
- Compiler/IDE: Code::Blocks / Turbo C++ / GCC
- Libraries: Standard C++ Libraries (iostream)

---

6. Algorithm / Logic

Step 1: Start the program.
Step 2: Initialize arrays for questions, options, and correct answers.
Step 3: Display the welcome message for the Quiz Application.
Step 4: For each question:

a) Display the question and four options (A, B, C, D).
b) Ask the user to input their answer.
c) Compare user input with the correct answer.
d) If the answer is correct, increment the score.
e) Display feedback (“Correct!” or “Wrong!”).

Step 5: After all questions are answered, display the final score.
Step 6: Show performance message based on the score (Excellent / Good / Keep Practicing).
Step 7: End the program.

---

7. Code Overview

Below is the main code used for implementing the Quiz Application:

```
#include <iostream>
using namespace std;

int main() {
    string questions[5] = {
        "1. What is the capital of India?",
        "2. Which data type is used to store decimal numbers?",
        "3. Who is known as the father of C++?",
        "4. Which operator is used for comparison?",
        "5. C++ is ____ programming language."
    };

    string options[5][4] = {
        {"A) Mumbai", "B) Delhi", "C) Kolkata", "D) Chennai"},
        {"A) int", "B) float", "C) char", "D) bool"},
        {"A) Bjarne Stroustrup", "B) James Gosling", "C) Dennis Ritchie", "D) Guido van Rossum"},
        {"A) ==", "B) =", "C) ++", "D) +"},
        {"A) Scripting", "B) Machine Level", "C) Object Oriented", "D) Markup"}
    };

    char correctAnswers[5] = {'B', 'B', 'A', 'A', 'C'};
    char userAnswer;
    int score = 0;

    cout << "\n*** QUIZ APPLICATION ***\n" << endl;

    for(int i = 0; i < 5; i++) {
```

```
    cout << questions[i] << endl;
    for(int j = 0; j < 4; j++) {
        cout << options[i][j] << endl;
    }

    cout << "Enter your answer (A/B/C/D): ";
    cin >> userAnswer;

    if(toupper(userAnswer) == correctAnswers[i]) {
        cout << "Correct! ✓\n\n";
        score++;
    } else {
        cout << "Wrong! ✗ Correct Answer: " << correctAnswers[i] << "\n\n";
    }
}

cout << "Your Final Score: " << score << "/5" << endl;

if(score == 5)
    cout << "Excellent! 😄" << endl;
else if(score >= 3)
    cout << "Good Work! 😳" << endl;
else
    cout << "Keep Practicing! 👍" << endl;

return 0;
}
```

8. Explanation of the Code

- **Array Declaration:**
  string questions[5] stores the quiz questions.
  string options[5][4] stores four options for each question. char correctAnswers[5] stores the correct option for each question.
- **Loop Control:**
  A for loop iterates through all 5 questions, displaying each one and checking the user's answer.
- **User Input:**
  The user inputs a character (A/B/C/D). The program uses toupper() to handle both uppercase and lowercase input.
- **Condition Checking:**
  If the user's answer matches the correct answer, the score is incremented.
- **Feedback System:**
  Displays messages such as *"Correct!"*, *"Wrong!"*, or *"Keep Practicing!"* depending on user performance.
- **Final Result:**
  After all questions, the final score and a motivational message are displayed.

---

10. Conclusion

The **Quiz Application** successfully demonstrates how to implement a simple quiz game using fundamental C++ concepts. The project helps in understanding **loops, conditionals, arrays, and input/output handling**. It also shows how to create an interactive program that provides real-time feedback and scoring.

This project can be enhanced in the future to include:

- Randomized questions
- Timers for each question
- Score storage in files
- Graphical user interface (GUI)

---

11. Learning Outcomes

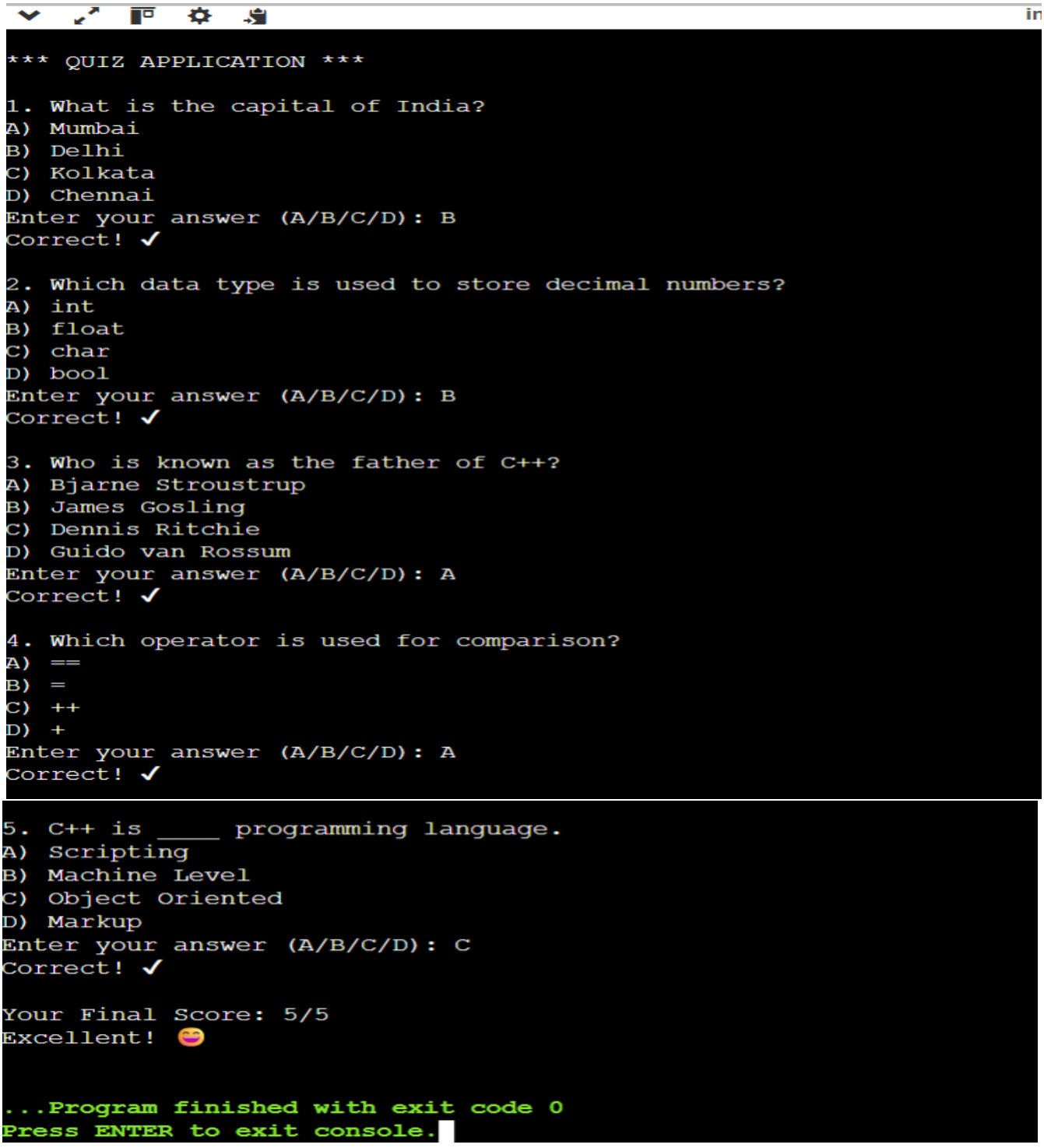
- Learned how to handle multiple-choice logic using arrays.
- Improved understanding of **loops**, **conditional statements**, and **data structures** in C++.
- Developed the ability to write structured, readable, and efficient code.
- Gained experience in interactive console application design.

---

12. References

- C++ Reference
- GeeksforGeeks – C++ Tutorials
- TutorialsPoint – C++ Programming Guide
- Classroom Notes and Faculty Materials

8. Screenshots / Output



CHANDIGARH
UNIVERSITY

Discover. Learn. Empower.