# **Answer for Part1:**

**Question1:DO NOT USE ARRAYS FOR THIS QUESTION**

**Provide a C++ for loop that repeats 8 times to displays the following 8 numbers as below. After the for loop, display**

**>>ANSWER:**

#include <iostream>

int main() {

int sum = 0;

for (int i = 1; i <= 8; i++) {

int number = i \* (i + 2);

std::cout << number << " ";

sum += number;

}

std::cout << "\nSum of these numbers is: " << sum << std::endl;

return 0;

}

**Question2:Suppose we define the following variables at the beginning of a program**

**const int LIMIT = 21000;**

**int result = 1;**

**int input = 1, previous = 1, counter = 0;**

**Write the while loop that lets the users to do the following:**

**\*Read an int value from the keyboard and assign it to the variable input**

**\*multiply input with result then then store back to the variable result.**

**\*The loop should iterate as long as result contains a value less than or equal LIMIT**

**\*count will be increased by 1 at every iteration**

**After the while loop stops, display message in the following format:**

**the output window is:**

**At time = 4**

**Previous result = 2184**

**LIMIT = 21000**

**Input = 15**

**Result = 32760**

**2B**

**If result = 21000 instead of result = 1 at the beginning.**

**How many times the while loop iterates such that result <= LIMIT**

**>>ANSWER:**

#include <iostream>

int main() {

const int LIMIT = 21000;

int result = 1;

int input = 1;

int previous = 1;

int counter = 0;

while (result <= LIMIT) {

std::cout << "Enter an integer: ";

std::cin >> input;

result \*= input;

counter++;

}

std::cout << "At time = " << counter << std::endl;

std::cout << "Previous result = " << result << std::endl;

std::cout << "LIMIT = " << LIMIT << std::endl;

std::cout << "Input = " << input << std::endl;

std::cout << "Result = " << result << std::endl;

return 0;

}

## **QUESTION 3:**

**3A**

**Suppose we define the following variables:**

**const int LIMIT = 21000;**

**int result = 1;**

**int input = 1, previous = 1, counter = 0;**

**Write the do..while loop that lets the users to do the following:**

**\*Read an int value from the keyboard and assign it to the variable input**

**\*multiply input with result then then store back to the variable result .**

**\*The loop should iterate as long as result contains a value less than or equal LIMIT**

**\*count will be increased by 1 at every iteration**

**After the while loop stops, display message in the following format, the output window is:**

**At time = 4**

**Previous result = 2184**

**LIMIT = 21000**

**Input = 15**

**Result = 32760**

**3B**

**If result = 21000 instead of result =1 at the beginning.**

**How many time the while loop iterates such that result <= LIMIT**

**>>ANSWER:  
IN C++:**

#include <iostream>

int main() {

const int LIMIT = 21000;

int result = 1;

int input = 1;

int counter = 0;

do {

std::cout << "Enter an integer: ";

std::cin >> input;

result \*= input;

counter++;

} while (result <= LIMIT);

std::cout << "At time = " << counter << std::endl;

std::cout << "Previous result = " << result << std::endl;

std::cout << "LIMIT = " << LIMIT << std::endl;

std::cout << "Input = " << input << std::endl;

std::cout << "Result = " << result << std::endl;

return 0;

}

**In Java:**import java.util.Scanner;

public class Main {

public static void main(String[] args) {

final int LIMIT = 21000;

int result = 1;

int input, previous = 1, counter = 0;

Scanner scanner = new Scanner(System.in);

do {

System.out.print("Enter an integer: ");

input = scanner.nextInt();

previous = result; // Keep track of the previous result

result \*= input;

counter++;

} while (result <= LIMIT);

System.out.println("At time = " + counter);

System.out.println("Previous result = " + previous);

System.out.println("LIMIT = " + LIMIT);

System.out.println("Input = " + input);

System.out.println("Result = " + result);

scanner.close(); // Close the scanner when done

}

}

## **QUESTION 4:**

**4A.**

**Write the code to do the following:**

**-Display message to ask then read input from the keyboard about the following information:**

**\* employeeID (string)**

**\* employeeName (string)**

**\* salary(float)**

**-open output file employee.txt to write:**

**-Write to the output file employee.txt the following information that have read input from the keyboard in the following format:**

**employeeID – employeeName – salary**

**For example:**

**1234567 - James Smith – 45000**

**-close file employee.txt**

**4B**

**Suppose in the file employee.txt already have 3 following lines:**

**1234567 – James Smith – 45000**

**2123312 – Mary Lane – 32500**

**1561175–Jonathan Nguyen– 51230**

**-provide the C++ code to open file employee.txt to read**

**-Read each line of the file then display on screen**

**-continue reading and displaying on the screen all the lines until end of the file**

**-Write: “End of the file employee.txt” on the screen**

**-close employee.txt file**

**>>ANSWER:**#include <iostream>

#include <fstream>

#include <string>

int main() {

// 4A: Read employee information and write to employee.txt

std::string employeeID, employeeName;

float salary;

std::cout << "Enter employee ID: ";

std::cin >> employeeID;

std::cout << "Enter employee name: ";

std::cin.ignore(); // Ignore the newline character left in the input buffer

std::getline(std::cin, employeeName);

std::cout << "Enter salary: ";

std::cin >> salary;

// Open the file for writing

std::ofstream outputFile("employee.txt");

if (!outputFile.is\_open()) {

std::cerr << "Failed to open the file for writing." << std::endl;

return 1; // Exit with an error code

}

// Write employee information to the file

outputFile << employeeID << " - " << employeeName << " - " << salary << std::endl;

// Close the file

outputFile.close();

// 4B: Read and display the contents of employee.txt

std::ifstream inputFile("employee.txt");

if (!inputFile.is\_open()) {

std::cerr << "Failed to open the file for reading." << std::endl;

return 1; // Exit with an error code

}

std::string line;

while (std::getline(inputFile, line)) {

std::cout << line << std::endl;

}

std::cout << "End of the file employee.txt" << std::endl;

// Close the file

inputFile.close();

return 0;

}

## **QUESTION 5:**

**Display the following menu:**

**MENU**

**1.Task 1**

**2.Task 2**

**3.Task 3**

**0. Exit**

**Type a number 1, 2, 3 to select a task or 0 to Exit:**

**Read the selected number from the keyboard.**

**Write the switch statement based on the selected number:**

**If selected number = 1: Display “Do task 1”**

**If selected number = 2: Display “Do task 2”**

**If selected number = 3: Display “Do task 3”**

**If selected number = 0: Display “Exit the menu”**

**For other number: Display “Invalid task, choose from 1, 2, 3 or 0”**

**After finishing one case either 1, 2, 3 or invalid selected number, you must redisplay the menu to allow users to select other task, until users select 0 to exit**

**>>ANSWER:**

#include <iostream>

int main() {

int choice;

do {

// Display the menu

std::cout << "MENU" << std::endl;

std::cout << "1. Task 1" << std::endl;

std::cout << "2. Task 2" << std::endl;

std::cout << "3. Task 3" << std::endl;

std::cout << "0. Exit" << std::endl;

std::cout << "Type a number (1, 2, 3) to select a task or 0 to Exit: ";

std::cin >> choice;

switch (choice) {

case 1:

std::cout << "Do task 1" << std::endl;

break;

case 2:

std::cout << "Do task 2" << std::endl;

break;

case 3:

std::cout << "Do task 3" << std::endl;

break;

case 0:

std::cout << "Exit the menu" << std::endl;

break;

default:

std::cout << "Invalid task, choose from 1, 2, 3, or 0" << std::endl;

break;

}

} while (choice != 0);

return 0;

}