Continuous Assessment - 01 Report

Only for Course Teacher						
		Needs Improvement	**Developing**	**Sufficient**	**Above Average**	**Total Mark**
Allocate mark & Percentage		**25%**	**50%**	**75%**	**100%**	**15**
Problem Analysis	**03**					
Solution Design	**02**					
Code Development	**06**					
Accuracy	**04**					
					Total obtained mark	
Comments						

Semester: Spring 2025

Student Name: Raktim Bain

Student ID: 242-35=560

Batch: 43 **Section:** M1

Course Code: SE 133 **Course Name:** Software Development Capstone Project

Course Teacher Name: Nazmus Sakib

Designation: Lecturer

Submission Date: 25/062025

Continuous Assessment 1: Individual Projects in C Programming

Task 1: Basic Calculator and Its Basic Operations

Code:

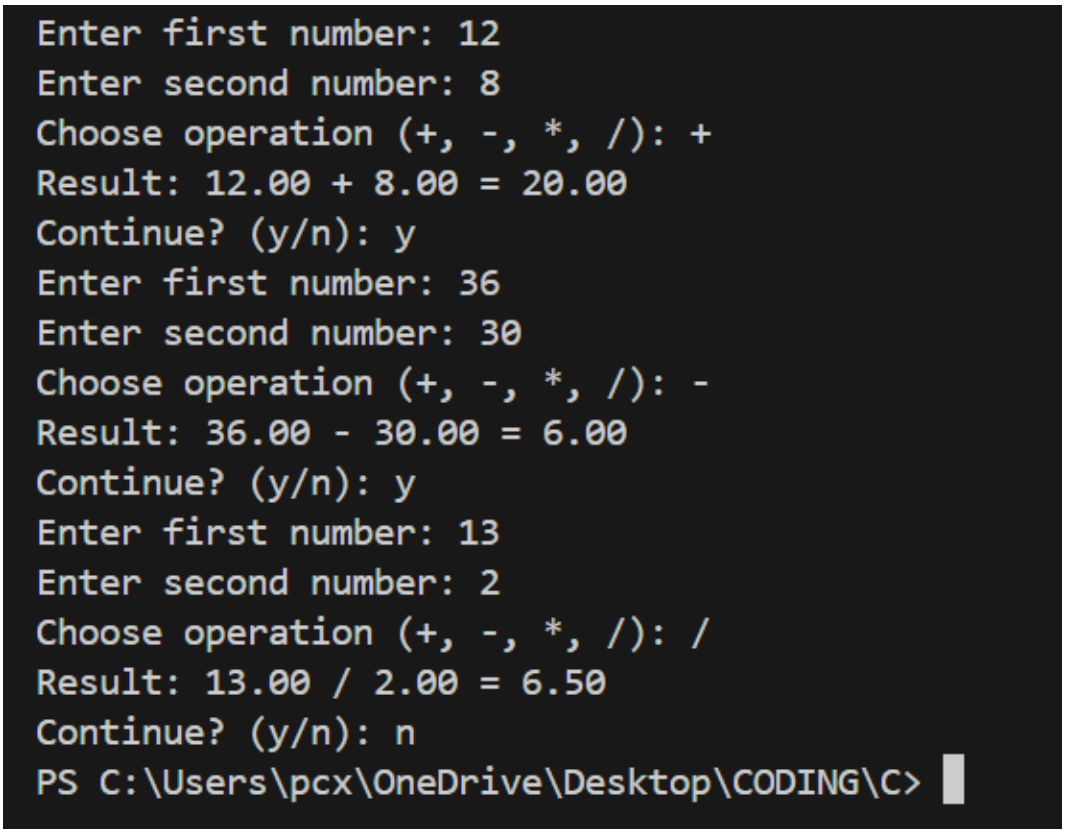
```c
#include <stdio.h>

int main() {
    double num1, num2, result;
    char op;
    char cont;

    do {
        printf("Enter first number: ");
        scanf("%lf", &num1);

        printf("Enter second number: ");
        scanf("%lf", &num2);

        printf("Choose operation (+, -, *, /): ");
        scanf(" %c", &op);

        switch (op)
        {
            case '+':
                result = num1 + num2;
                printf("Result: %.2lf + %.2lf = %.2lf\n",
num1, num2, result);
```

```
                break;
            case '-':
                result = num1 - num2;
                printf("Result: %.2lf - %.2lf = %.2lf\n",
num1, num2, result);
                break;
            case '*':
                result = num1 * num2;
                printf("Result: %.2lf * %.2lf = %.2lf\n",
num1, num2, result);
                break;
            case '/':
                if (num2 == 0) {
                    printf("Error: Division by zero!\n");
                } else {
                    result = num1 / num2;
                    printf("Result: %.2lf / %.2lf =
%.2lf\n", num1, num2, result);
                }
                break;
            default:
                printf("Invalid operation!\n");
        }

        printf("Continue? (y/n): ");
        scanf(" %c", &cont);

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

    return 0;
}
```

Program Explanation:

This C program implements a **simple calculator** that performs **basic arithmetic operations** (+, -, *, /) on two numbers. The user is prompted to enter two numbers and an operation. After displaying the result, the user is asked if they want to perform another calculation. The process repeats as long as the user enters 'y' or 'Y'.

Output:



Task 2: Compare Computer-Generated and User-Defined Numbers

Code:

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main() {
    int userNum, compNum;
    char tryAgain;

    // Initialize random number generator
    srand(time(0));

    do {
        do {
            printf("Enter a number (1-100): ");
            scanf("%d", &userNum);
            if (userNum < 1 || userNum > 100) {
                printf("Error: Number out of range!\n");
            }
        } while (userNum < 1 || userNum > 100);

        compNum = (rand() % 100) + 1;
        printf("Computer-generated number: %d\n",
compNum);

        if (compNum > userNum) {
            printf("%d is greater than %d.\n", compNum,
userNum);
        } else if (compNum < userNum) {
```

```
            printf("%d is less than %d.\n", compNum,
userNum);
        } else {
            printf("Numbers are equal.\n");
        }

        if (compNum % 2 == 0) {
            printf("%d is even.\n", compNum);
        } else {
            printf("%d is odd.\n", compNum);
        }

        printf("Try again? (y/n): ");
        scanf(" %c", &tryAgain);

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

    return 0;
}
```

Program Explanation: Number Guessing with Random Generation

This C program lets the user enter a number between **1 and 100**, then compares it with a randomly generated number by the computer. It provides feedback about the comparison, whether the computer's number is **even or odd**, and lets the user choose to repeat the process.

Output:

```
Enter a number (1-100): 109
Error: Number out of range!
Enter a number (1-100): 78
Computer-generated number: 49
49 is less than 78.
49 is odd.
Try again? (y/n): y
Enter a number (1-100): 29
Computer-generated number: 48
48 is greater than 29.
48 is even.
Try again? (y/n): n
PS C:\Users\pcx\OneDrive\Desktop\CODING\C>
```

Task 3: Checking Name, ID, Grade, and CGPA of Multiple Students

Code:

```c
#include <stdio.h>
#include <string.h>

struct Student
{
    char name[50];
    char id[20];
    char grade;
    float cgpa;
};

int main()
{
    int n;
```

```c
    printf("Enter number of students: ");
    scanf("%d", &n);

    struct Student students[n];

    // Input student information
    for (int i = 0; i < n; i++)
    {
        printf("Student %d:\n", i + 1);
        printf("Enter Name: ");
        scanf(" %[^\n]", students[i].name);
        printf("Enter ID: ");
        scanf("%s", students[i].id);
        printf("Enter Grade (A-F): ");
        scanf(" %c", &students[i].grade);
        printf("Enter CGPA (0.0-4.0): ");
        scanf("%f", &students[i].cgpa);
    }

    // Display all student info
    printf("\n=== Student Information ===\n");
    printf("Name\tID\tGrade\tCGPA\n");

    for (int i = 0; i < n; i++)
    {
        printf("%s\t%s\t%c\t%.2f\n", students[i].name,
students[i].id, students[i].grade, students[i].cgpa);
    }

    // Search loop
    char searchID[20];
    while (1) {
        printf("\nEnter ID to search (or 'exit'): ");
```

```c
        scanf("%s", searchID);

        if (strcmp(searchID, "exit") == 0)
        {
            break;
        }

        int found = 0;
        for (int i = 0; i < n; i++)
        {
            if (strcmp(searchID, students[i].id) == 0) {
                printf("Found: %s, ID: %s, Grade: %c,
CGPA: %.2f\n",
                       students[i].name, students[i].id,
students[i].grade, students[i].cgpa);
                found = 1;
                break;
            }
        }

        if (!found)
        {
            printf("Student with ID %s not found.\n",
searchID);
        }
    }

    return 0;
}
```

Program Explanation: Student Information System with Search

This C program allows the user to:

1. **Input student information** (Name, ID, Grade, CGPA).
2. **Display all student records**.
3. **Search for a student by ID** repeatedly until the user types "exit".

Output:

```
Enter number of students: 3
Student 1:
Enter Name: Jimmy
Enter ID: 001
Enter Grade (A-F): A
Enter CGPA (0.0-4.0): 4
Student 2:
Enter Name: Chadlar
Enter ID: 121
Enter Grade (A-F): A
Enter CGPA (0.0-4.0): 3.78
Student 3:
Enter Name: Karl
Enter ID: 222
Enter Grade (A-F): D
Enter CGPA (0.0-4.0): 1.26

=== Student Information ===
Name    ID      Grade   CGPA
Jimmy   001     A       4.00
Chadlar 121     A       3.78
Karl    222     D       1.26

Enter ID to search (or 'exit'): exit
PS C:\Users\pcx\OneDrive\Desktop\CODING\C>
```