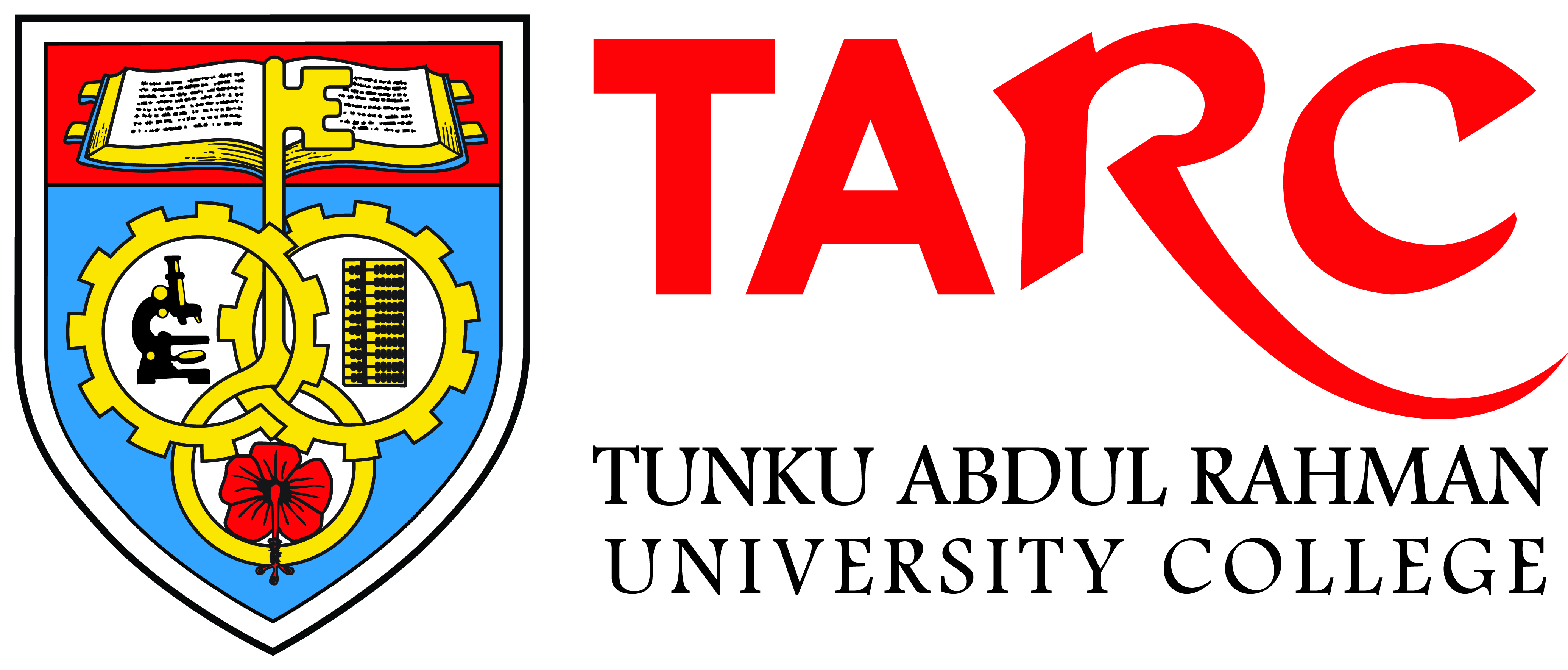
****

TUNKU ABDUL RAHMAN UNIVERSITY COLLEGE

FACULTY OF COMPUTING AND INFORMATION TECHNOLOGY

SEMESTER 1 - ACADEMIC YEAR 2020/2021

**AACS1074**

**PROGRAMMING CONCEPTS & DESIGN I**

**(ASSIGNMENT)**

**STUDENT NAME : Lee Khoon Hong**

**STUDENT ID :**

**PROGRAMME :**  **DCO**

**TUTORIAL GROUP :**  **G1**

**TUTOR :** **Hooi Leng Kheoh**

**DATE RECEIVED :**

**Assessment Criteria & Feedback Form** AACS1074 PCD I Assignment

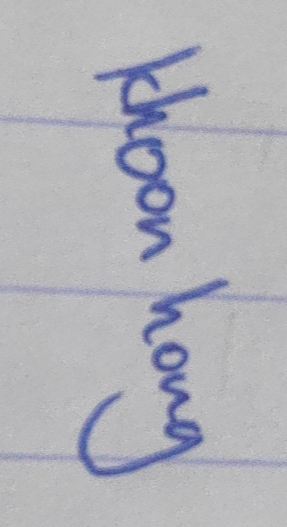
|  |  |  |  |
| --- | --- | --- | --- |
| **Program (75%)** |  | **Report (25%)** |  |
| A - Submitted C Code, shows understanding **( 5)** |  | A - Submitted Var & Const, f/chart OR **(5)**  Pseudocode, screenshots |  |
| B - Submitted C Code, shows understanding **( 5)** |  | B - Submitted Var & Const, f/chart OR **(5)**  Pseudocode, screenshots |  |
| 1. Basic calculations for different details **(15)**    * Correct inputting values    * Correct processing / calculation    * Correct output values / totals    * Correct output / report I format    * Used at least 4 constants |  | Intro/ description **(1)**   * Clarity / Useful info |  |
| Added Features @ 2 **(4)**   * Clear description / Usefulness |  |
| 1. Use conditional statement to display appropriate messages and reports required  **(10)**    * If-statement : new cases +/- msg    * If-statement : choose report type    * Correct table totals & changes    * Correct output / report II format    * General - constants etc |  | Overall Program Design **(2)**   * Structure Chart, >= 3 level * Correct / Suitable / Tidy |  |
| Method of Solution **(2)**   * Flowchart OR pseudocode * Correct use, eg. Symbols, indentation etc |  |
| 1. Use loop to process data for more than 1 day **(14)**    * Appropriate looping    * Correct total days & to/from day nos    * Highest & lowest cases + day nos    * Suitable ending message |  | Constants & Variables **(1)**   * Correct list pasted from program * table for constants   + Name, value, purpose * table for variables   + Name, data type, purpose |  |
| Added Features **(6)**   * 2 extra features @ 3 marks   o Usefulness to user/customer o Complexity of program/logic o Successfully implemented   * NOT considered as features - use of program constructs / commands (eg. Blinking screen) |  | (Screenshots showing the added features must be included somewhere in the chapter, and indicated)  Outputs - Run 1 **(1)**   * description * Scenario table (new page) * Complete set of screenshots |  |
| User Friendliness **(4)**   * Prompts & Responses * General courtesy * Overall Screen Layout is tidy |  | Outputs - Run 2 **(2)**   * description * Scenario table (new page) * Complete set of screenshots |  |
| Readability **(10)**   * suitable identifier names * constants used as appropriate * adequate & useful comments * program indentation * clear programming constructs |  | Outputs - Run 3 **(2)**   * description * Scenario table (new page) * Complete set of screenshots |  |
| Originality/Creativity **(6)**   * uniqueness * creativity |  | Late Penalty   * less 10% per day |  |
| Penalty (if any)  **(@ -2)**   * Inefficient / illogical structures, eg:   + unnecessary looping   + inappropriate structures * -2 per occurrence (at tutor's discretion) * Late submission for A, B, C @ -2 per day |  | Other Penalty (if any) **(@ -1) / ( -5)**   * Missing / incomplete item/chapter   + -1 per item * Flowchart/pseudocode is totally different from program   + -5 (at tutor's discretion) |  |
|  |  |  |  |
| **Total =** | | |  |

**Declaration of Originality**

I declare that this assignment is free from all forms of plagiarism and for all intents and purposes is my own work. I understand that I will be penalized if I have not complied with TAR UC’s Plagiarism policy.



<Insert your photo>

Student Name : \_\_\_\_\_Lee Khoon Hong\_\_\_\_\_\_\_

Signature :

Date : \_\_\_\_\_31/7/2020\_\_\_\_\_\_\_\_\_\_\_\_\_

**Table of Content**

|  |  |  |
| --- | --- | --- |
| **No.** | **Title** | **Page No.** |
| 1 | Brief description / purpose |  |
| 2 | Overall program structure |  |
| 3 | Method of solution |  |
| 4 | Added features |  |
| 5 | Program testing & outputs |  |
| 6 | Constant & variables |  |

1. **Brief Description / Purpose**

< Maximum 2 pages. Explain (eg. to a stranger) what your system/program is about. Who uses it, and for what purpose? What can it do, etc>

1. **Overall Program Design**

< 1 page - structure chart drawn in 3 or more levels >

1. **Method of Solution**

< flowchart(s) OR pseudocode – do not need to show detail steps for generating splash screen >

1. **Added Features**

< For each feature:

Describe it.

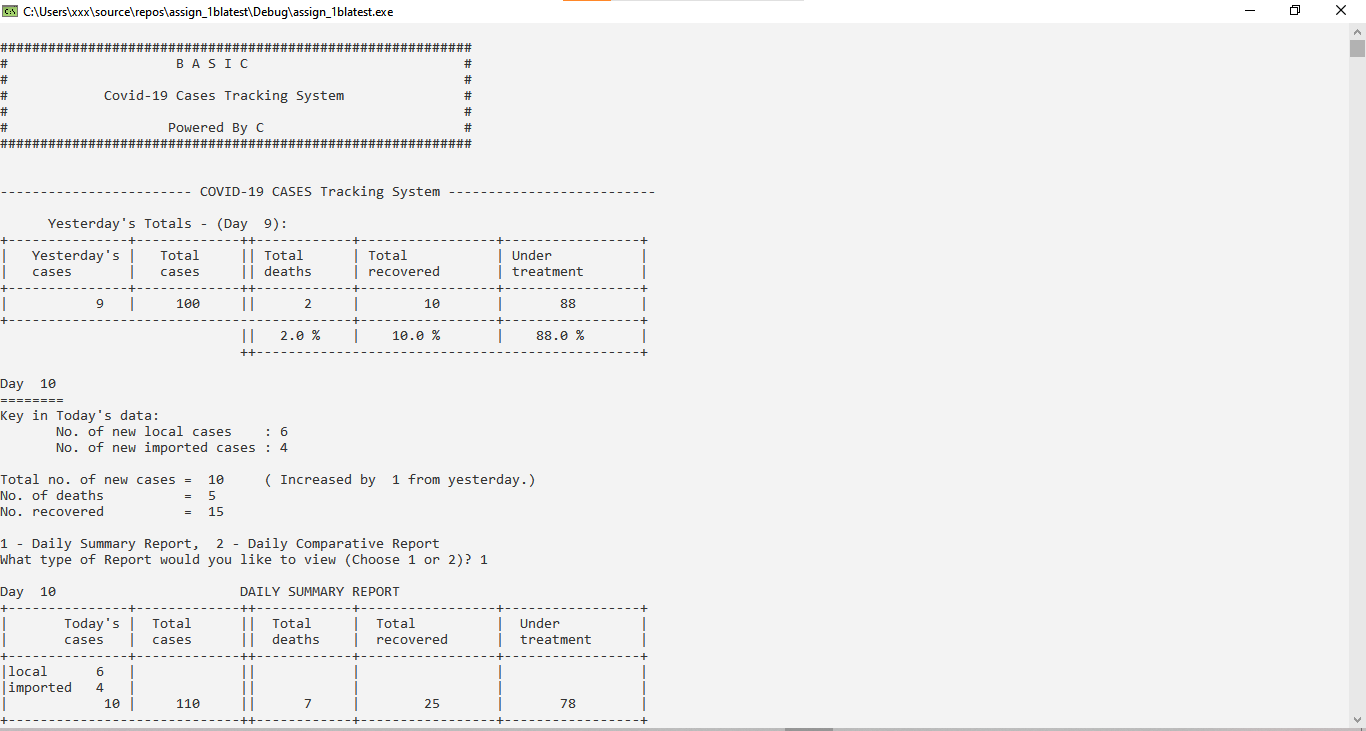
Explain why it is useful.

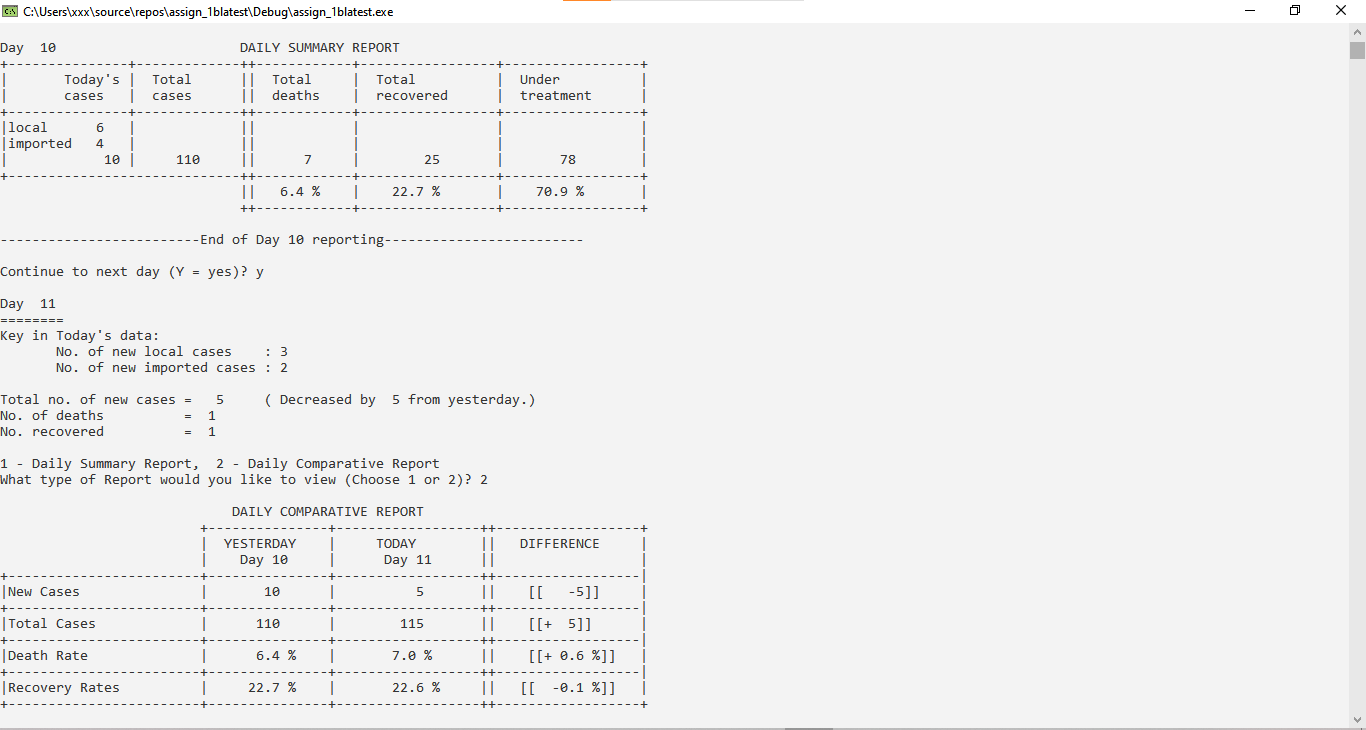
Explain how you implemented it in your program. >

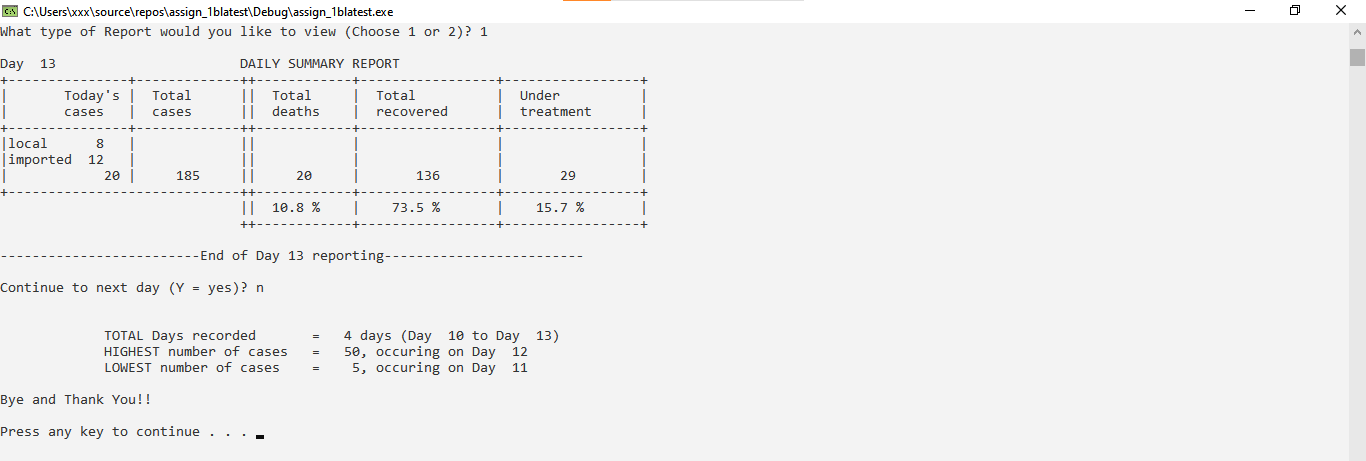
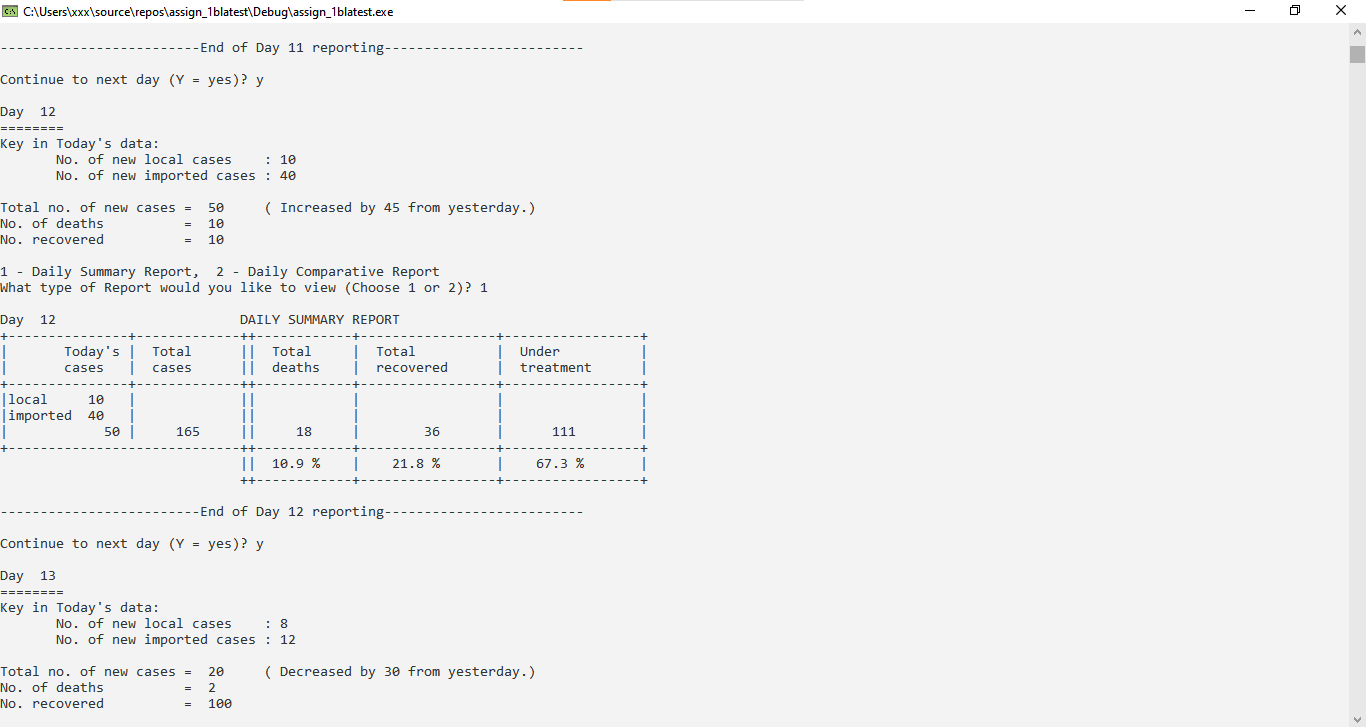
< etc >

1. **Program Testing & Outputs** 
   1. **Run 1 Scenario -**

**Test Data + Expected Outputs**







* 1. **Run 2 Scenario -**

**Test Data + Expected Outputs**

* 1. **Run 3 Scenario -**

**Test Data + Expected Outputs**

1. **Constants & variables**

**Program Code:**

Pre-processor

#include<stdlib.h>

#include<stdio.h>

#include<math.h>

#pragma warning (disable: 4996)

#define MAX\_SIZE 100

Variables and arrays

double previousNewCases = 0, previousTotalCases = 0, previousTotalDeaths = 0, previousTotalRecovered = 0, previousUnderTreatment = 0;

double currentNewCases, currentLocal, currentImport, currentTotalCases, currentTotalDeaths, currentTotalRecovered, currentUnderTreatment, previousDeathRates, previousRecoveryRates, currentRecoveryRates, currentDeathRates, dayToday = 10, dayYesterday = 9;

double ratesCurrentTotalDeaths, ratesCurrentTotalRecovered, ratesCurrentUnderTreatment, ratesPreviousTotalDeaths, ratesPreviousTotalRecovered, ratesPreviousUnderTreatment, differenceNewCases, differenceTotalCases, differenceDeathRates, differenceRecoveryRates, totalDaysRecorded, highestNumOfCases, dayHighestNumOfCases, lowestNumOfCases, dayLowestNumOfCases;

int userDigitPrompt, i; char userYesNoPrompt; int arrayCount = 0;

int cases[MAX\_SIZE], duplicateCases[MAX\_SIZE];

Formulas

ratesPreviousTotalDeaths = (DAY9TOTALDEATHS / DAY9TOTALCASES) \* 100;

ratesPreviousTotalRecovered = (DAY9TOTALRECOVERED / DAY9TOTALCASES) \* 100;

ratesPreviousUnderTreatment = (DAY9UNDERTREATMENT / DAY9TOTALCASES) \* 100;

cases[arrayCount] = DAY9NEWCASES; duplicateCases[arrayCount] = DAY9NEWCASES;

currentTotalCases = DAY9TOTALCASES + currentNewCases;

currentTotalDeaths += DAY9TOTALDEATHS;

currentTotalRecovered += DAY9TOTALRECOVERED;

currentUnderTreatment = currentTotalCases - currentTotalDeaths - currentTotalRecovered;

ratesCurrentTotalDeaths = (currentTotalDeaths / currentTotalCases) \* 100;

ratesCurrentTotalRecovered = (currentTotalRecovered / currentTotalCases) \* 100;

ratesCurrentUnderTreatment = (currentUnderTreatment / currentTotalCases) \* 100;

previousDeathRates = (DAY9TOTALDEATHS / DAY9TOTALCASES) \* 100;

previousRecoveryRates = (DAY9TOTALRECOVERED / DAY9TOTALCASES) \* 100;

currentDeathRates = (currentTotalDeaths / currentTotalCases) \* 100;

currentRecoveryRates = (currentTotalRecovered / currentTotalCases) \* 100;

differenceNewCases = currentNewCases - DAY9NEWCASES;

differenceTotalCases = currentTotalCases - DAY9TOTALCASES;

differenceDeathRates = currentDeathRates - previousDeathRates;

differenceRecoveryRates = currentRecoveryRates - previousRecoveryRates;

**Constants variable:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | constant | values | purpose |
| 1. | DAY9NEWCASES | 9 | It stores the number of new cases on Day 9 |
| 2. | DAY9TOTALCASES | 100 | It stores the number of total cases on Day 9 |
| 3. | DAY9TOTALDEATHS | 2 | It stores the number of total deaths on Day 9 |
| 4. | DAY9TOTALRECOVERED | 10 | It stores the number of total recovered on Day 9 |
| 5. | DAY9UNDERTREATMENT | 88 | It stores the value of the total number of people who are under recovery that are recorded from the previous days on Day 9 |
| 6. | STARTINGDAY | 10 | To calculate the total days recorded by the program |

**Variables**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Variable | Data type | Purpose |
| 1. | previousNewCases | Double | This variable stores the number of previous day new added cases. |
| 2. | previousTotalCases | Double | This variable stores the value of previous day total cases. |
| 3. | previousTotalDeaths | Double | This variable stores the value of previous day total deaths. |
| 4. | previousTotalRecovered | Double | This variable stores the value of previous day total recovered patients. |
| 5. | previousUnderTreatment | Double | This variable stores the value of previous day under-treatment patients. |
| 6. | currentNewCases | Double | This variable stores the value of current day newly added cases. (current day local cases + current day imported cases) |
| 7. | currentLocal | Double | This variable stores the value of current day local cases. (prompt user input) |
| 8. | currentImport | Double | This variable stores the value of current day imported cases. (prompt user input) |
| 9. | currentTotalCases | Double | This variable stores the value of current day total cases. (previous day total cases + current day new cases) |
| 10. | currentTotalDeaths | Double | This variable stores the value of current day total death. (previous day total deaths + current day total deaths) |
| 11. | currentTotalRecovered | Double | This variable stores the value of current day total recovered patients. (current day total recovered + previous day total recovered) |
| 12. | currentUnderTreatment | Double | This variable stores the value of current day under-treatment patients. (current day under-treatment + previous day under-treatment) |
| 13. | previousDeathRates | Double | This variable stores the value of previous day death rates. (previous day total deaths divide by previous day total cases and then finally divide by 100) |
| 14. | previousRecoveryRates | Double | This variable stores the value of previous day recovery rates. (previous day total recovered divide by previous day total cases and then finally divide by 100) |
| 15. | currentRecoveryRates | Double | This variable stores the value of current day recovery rates. (current day total recovered divide by current day total cases and then finally divide by 100) |
| 16. | currentDeathRates | Double | This variable stores the value of current day death rates. (current day total deaths divide by current day total cases and then finally divide by 100) |
| 17. | dayToday | Double | This variable stores the value of the count of the day value today, which initially starts from day 10. (dayToday++) |
| 18. | dayYesterday | Double | This variable stores the value of the count of the day value yesterday, which initially starts from day 9. (dayYesterday++) |
| 19. | ratesCurrentTotalDeaths | Double | This variable stores the value of current day total death rates. (current day total deaths divide by current day total cases and then finally divide by 100) |
| 20. | ratesCurrentTotalRecovered | Double | This variable stores the value of current day total recovered rates. (current day total recovered divide by current day total cases and then finally divide by 100) |
| 21. | ratesCurrentUnderTreatment | Double | This variable stores the value of current day under-treatment rates. (current day under-treatment divide by current day total cases and then finally divide by 100) |
| 22. | ratesPreviousTotalDeaths | Double | This variable stores the value of previous day total death rates. (previous day total deaths divide by previous day total cases and then finally divide by 100) |
| 23. | ratesPreviousTotalRecovered | Double | This variable stores the value of previous day total recovered rates. (previous day total recovered divide by previous day total cases and then finally divide by 100) |
| 24. | ratesPreviousUnderTreatment | Double | This variable stores the value of previous day under-treatment rates. (previous day under-treatment divide by previous day total cases and then finally divide by 100) |
| 25. | differenceNewCases | Double | This variable stores the value of the difference of new cases. (current day new cases – previous day new cases) |
| 26. | differenceTotalCases | Double | This variable stores the value of the difference of total cases. (current day total cases – previous day total cases) |
| 27. | differenceDeathRates | Double | This variable stores the value of the difference of death rates. (current day death rates – previous day death rates) |
| 28. | differenceRecoveryRates | Double | This variable stores the value of the difference of recovery rates. (current day recovery rates – previous day recovery rates) |
| 29. | totalDaysRecorded | Double | This variable stores the value of the total number of days the user input. |
| 30. | highestNumOfCases | Double | This variable stores the value of the highest number of cases ever recorded in the program. |
| 31. | dayHighestNumOfCases | Double | This variable stores the value of the day of highest number of cases ever recorded in the program. |
| 32. | lowestNumOfCases | Double | This variable stores the value of the lowest number of cases ever recorded in the program. |
| 33. | dayLowestNumOfCases | Double | This variable stores the value of the day of lowest number of cases ever recorded in the program. |
| 34. | userDigitPrompt | Int | This variable stores the value of ‘1’ and ‘2’, which leads to different output after user entered their desired input value. |
| 35. | i | Int | This variable stores the value of for loop variable looping starting variable. |
| 36. | userYesNoPrompt | Char | This variable stores the value of char ‘y’ or char ‘n’, which the user would have to input while prompted. |
| 37. | arrayCount | Int | This variable stores the value of count of the array, which is initiated while calling a particular array in the program. |