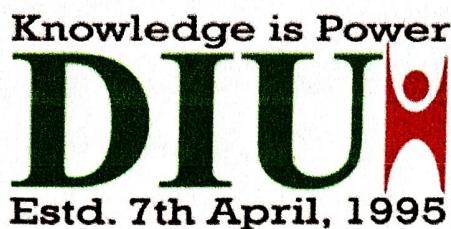


Dhaka International University



DEPARTMENT OF CSE

LAB REPORT

COURSE NAME : Structured Programming Language Lab

COURSE CODE : 0613 - 102

REPORT NO :

REPORT ON : C Programs for Conditional Statement –
Vowel/consonant, Pass/Fail, Grading system

<u>SUBMITTED BY</u>	<u>SUBMITTED TO</u>
NAME : Mir Yeasin Abraan ROLL : 35 REG. NO : CS-D-98-23-12T358 BATCH : 98 SEMESTER : 1st	Rakib Mahmud Lecturer Dept. of CSE, DIU

DATE OF SUBMISSION :

Date of Performance : 14 September 2024

Lab Report: C Programs Using Conditional Statements - vowel or consonant, Pass or Fail, and Grading system.

Objectives:

- Understand how to use conditional statements - if-else and if-else if in C programming.
- Implement programs to:
 - Identify vowels or consonant
 - Determine if a student has passed or failed based on their marks.
 - Assign grades based on marks using a grading scale.

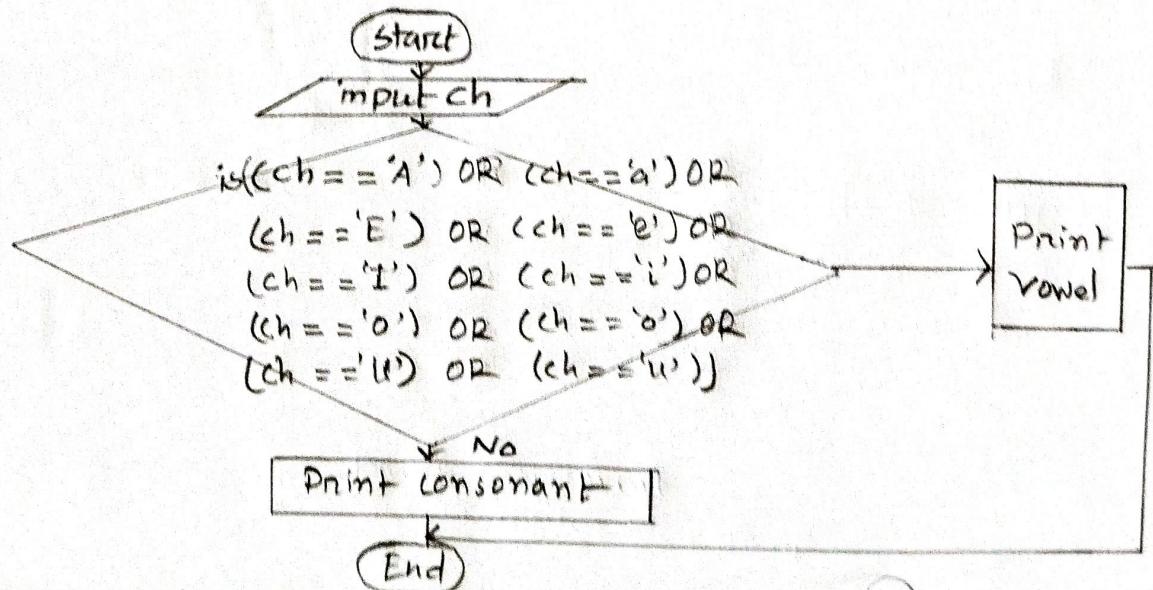
Introduction:

Each of these programs uses conditional statements to make decisions based on the user's input.

Vowel or Consonant Program:

Explanation: The program checks if an input character is a vowel (a, e, i, o, u both uppercase and lowercase) or a consonant using conditional statements.

Flowchart:



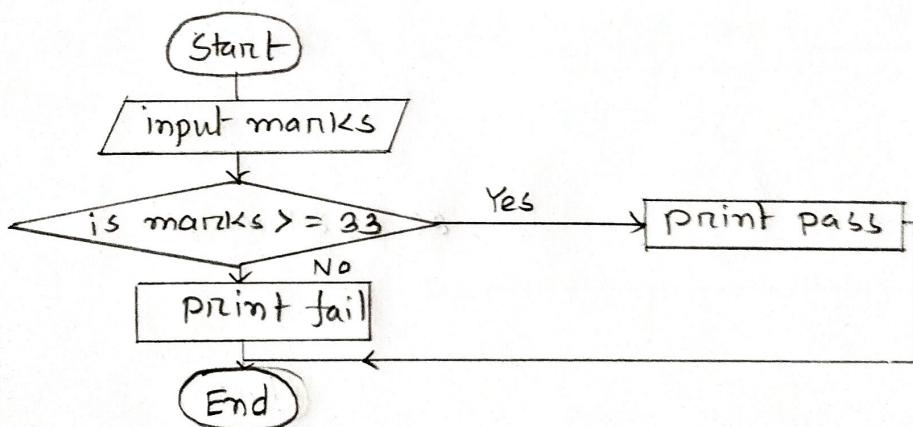
Algorithm:

1. Start
2. input ch
3. [S (ch == 'A') OR (ch == 'a') OR (ch == 'E') OR (ch == 'e')
OR (ch == 'I') OR (ch == 'i') OR (ch == 'O') OR (ch == 'o')
OR (ch == 'U') OR (ch == 'u')]:
 - i. Yes, print vowel
 - ii. No, print consonant
4. End.

• Pass or Fail Program:

... Explanation: This program checks if the marks entered by the user are greater than or equal to 33. If yes, the student passes; otherwise, they fail.

... Flowchart:



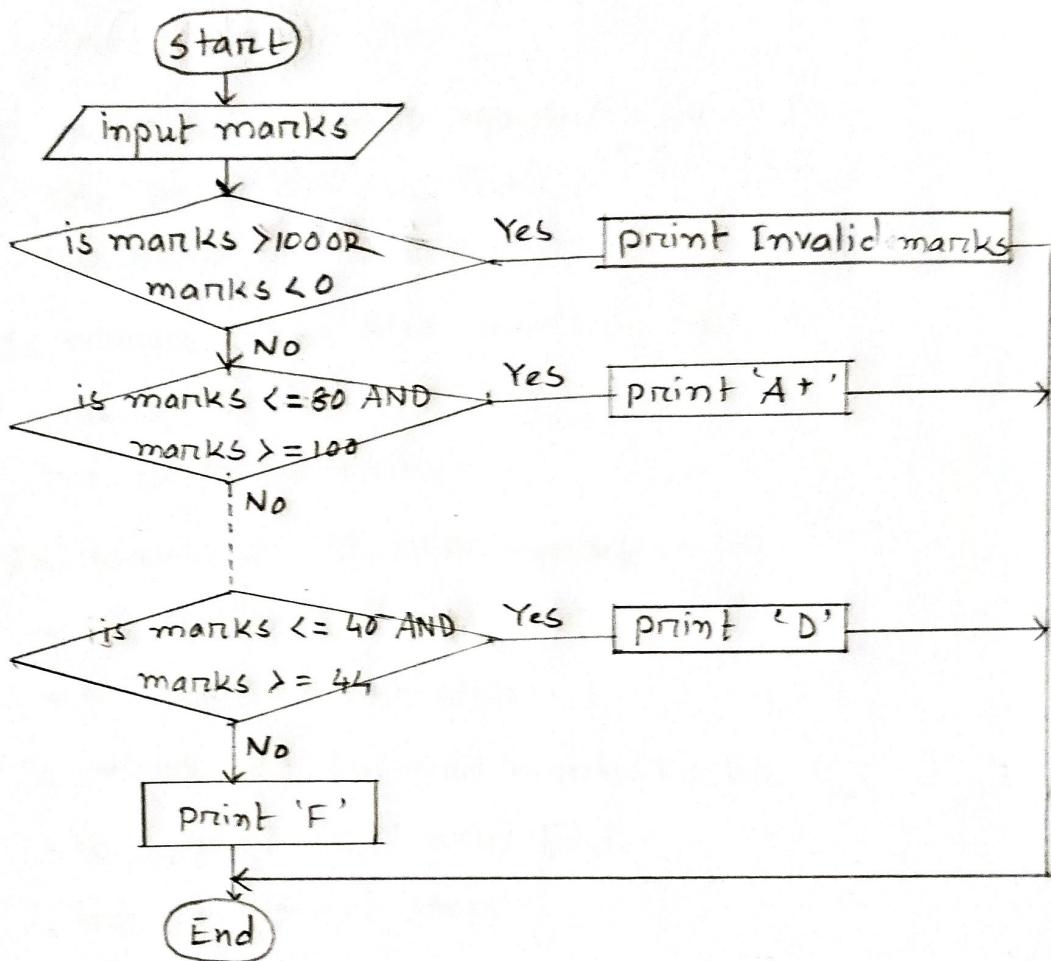
... Algorithm:

1. Start
2. Input marks
3. Is marks ≥ 33
 - i. Yes, print pass
 - ii. No, print fail
4. End

- Grading system program:

- Explanation: The program assigns a grade based on the input marks using a multi-tiered grading system, with different grades from A+ to F.

- Flowchart:



- Algorithm:

- start
- input marks
- Is $(\text{marks} > 100) \text{ OR } (\text{marks} < 0)$
 - Yes, print Invalid marks and End
 - No, go to next step
- Is $\text{marks} \leq 100 \text{ AND } \text{marks} \geq 80$
 - Yes, print A+ and End
 - No, go to next step
- Is $\text{marks} \leq 40 \text{ AND } \text{marks} \geq 44$
 - Yes, print D and End
 - No, print F and End

- Is marks ≤ 79 AND marks ≥ 75
 - Yes, print 'A' and End
 - No, go to next step
- Is marks ≤ 74 AND marks ≥ 70
 - Yes, print 'A-' and End
 - No, go to next step
- Is marks ≤ 69 AND marks ≥ 65
 - Yes, print 'B+' and End
 - No, go to next step
- Is marks ≤ 64 AND marks ≥ 60
 - Yes, print 'B' and End
 - No, go to next step
- Is marks ≤ 59 AND marks ≥ 55
 - Yes, print 'B-' and End
 - No, go to next step
- Is marks ≤ 54 And marks ≥ 50
 - Yes, print 'C+' and End
 - No, go to next step
- Is marks ≤ 49 AND marks ≥ 45
 - Yes, print 'C' and End
 - No, go to next step
- Is marks ≤ 44 AND marks ≥ 40
 - Yes print 'D' and End
 - No, go to next step
- Otherwise
 - Yes, print 'F' and end

Discussion:

In this section we will learn about the 'Presentation of code', 'Output' and 'Explanation of the code'. The shown code will be as in 'IDE' and the output will be as in console.

⇒ Vowel or Consonant program

... code :

```
#include <stdio.h>
int main()
{
    char ch;
    printf("Enter an alphabat : ");
    scanf("%c", &ch);
    if(ch == 'A' || ch == 'a' || ch == 'E' || ch == 'e' ||
       ch == 'I' || ch == 'i' || ch == 'O' || ch == 'o' ||
       ch == 'U' || ch == 'u')
        printf("%c is a vowel\n", ch);
    else
        printf("%c is a consonant\n", ch);
    return 0;
}
```

... output :

Enter an alphabat : a
a is a vowel

... Explanation :

- #include <stdio.h> : For input output operations.
- ch : variable, for store value
- printf() : Prints output in the console.
- scanf() : Gets input from user
- if-else : conditional statement for checking condition
- return 0 : returns to main function.

⇒ Pass or Fail Program:

... code:

```
#include <stdio.h>
int main () {
    int marks,
        printf ("Enter marks : ");
    scanf ("%d", &marks);
    if (marks >= 33)
        {printf ("Pass \n");}
    else
        {printf ("Fail \n");}
    return 0;
}
```

... output:

Enter marks: 89

Pass

... Explanation:

- #include <stdio.h>: For input-output operations.
- marks: Variable to store value
- printf(): Prints output in the console
- scanf(): Gets input from the user
- if-else: Conditional statement for decision making.
- return 0; Returns to main function.

⇒ Grading System Program.

... code:

```
#include <stdio.h>
int main () {
    int marks;
    printf ("Enter your marks : ");
    scanf ("%d", &marks);
```

```
if(marks < 0 || marks > 100)
{ printf("Invalid marks"); }

else if(marks) = 80 && marks <= 100)
{ printf("A+"); }

else if(marks) >= 75 && marks <= 79 )
{ printf("A"); }

elseif(marks) >= 70 && marks <= 74 )
{ printf("A-\n"); }

else if(marks) >= 65 && marks <= 69)
{ printf("B+\n"); }

else if(marks) = 60 && marks <= 64)
{ printf("B\n"); }

elseif(marks) >= 55 && marks <= 59)
{ printf("B-\n"); }

elseif(marks) >= 50 && marks <= 54)
{ printf("C+\n"); }

elseif(marks) >= 45 && marks <= 49)
{ printf("C\n"); }

elseif(marks) >= 40 && marks <= 44)
{ printf("D\n"); }

else
{ printf("F\n"); }

return 0;
```

... Explanation :

- `#include <stdio.h>` : For input/output operations
- `marks` : variable for storing value.
- `printf()` : prints output in the console
- `scanf()` : gets input from user
- `if-else if` : conditional statement for decision making
- `return 0` : returns to main function.

Conclusion :

In this section we will discuss about the challenges we encountered as 'limitations' and the use of these program as 'Applications.'

... Limitations :

- Single Character Input : The vowel/consonant program only works for a single character input and doesn't handle non-alphabetic characters.
- No Range Validation : The grading program does not handle where marks exceed 100 or are negative.
- Static Threshold for pass/fail : The pass/fail program uses a fixed threshold (33), which may vary by institution or country.

... Applications :

- Vowel/Consonant Program : Can be used in text processing, language learning applications, or spelling correction systems.
- Pass/Fail Program : Can be used in academic systems to

determine whether a student passes a subject.

- Grading Program : Useful in educational applications for automating grade assignment based on examination results.

References:

- C Standard Library Documentation
- learn.microsoft.com
- github.com
- openai.com