EXAM CALCULATOR

INTRODUCTION

In many university programs, courses typically feature a standardized assessment structure, involving either **two or three exams** during the course of a single semester. Under the two-exam structure, students are evaluated through a **mid-term examination, which constitutes 40%** of their **final grade, and a comprehensive final exam that carries a weight of 60%.** Alternatively, in the three-exam approach, students face three distinct assessments, with **the first and second mid-term exams each contributing 30% to the final grade, and a final exam encompassing 40% of the overall assessment.** This assessment framework allows for a comprehensive evaluation of students' knowledge and understanding of the course material, fostering a balanced approach to their academic progress.

INSTRUCTIONS

1) Create a menu like left side that involves selection two or three exam structure selection and press to the screen**(10pts)**

```
Please Select Exam Structure
1 – Two Exam Model: Midterm, and Final
2 – Three Exam Model: Midterm1, Midterm2, and Final

Please enter your selection:
```

2) First, You need to read user's selection. **Scanf** function is good choice! But you are free to use **getchar() (10pts)**

3) Now, You need to write first part of exam structure. If user pressed to **"1"** You need to accept two inputs form user and take their percentages %40, and %60 sequentially. The output screen has been shown on the left. **(35pts)**

```
Please enter your selection:1

Please Enter Midterm Score:30
Please Enter Final Score:60
Your grade is: 48.000000
```

4) Another side of program, if the user pressed to **"2"** You need to accept three inputs form user and take their percentages %30, %30, and %40 sequentially. The output screen has been shown on the left.**(40pts)**

```
Please enter your selection:2

Please Enter Midterm1 Score:30
Please Enter Midterm2 Score:40
Please Enter Final Score:50
Your grade is: 41.000000
```

5) Be careful that the result has been shown in float type! So you need to use type casting. I am giving an example for how to use **type casting in C language** at below. **(5pts)**

```c
int a = 8;
int b = 20;
int c = 30;
float MyResult = 0;

MyResult = (float)((float)(b/a) + (float)c);
```

Important: You are free to solve all tasks whatever you want. If, while, for, switch-case, etc. you may use.