Mini Project: Learning Aid Software

Project Name: Newton's Law of Motion
Instructor's name: Dr. Mohd Shahrin Bin Abu Hanifah

Name	Matric number	Task
Sufi Hakim Bin Sulaiman (Leader)	2012507	Report and video editing.
Muhammad Azim Iskandar Bin Mohd Sofi	2010613	Coding software based on flowcharts and ideas.
Ikmal Ikhwan Bin Mohamad Rafaii	2013043	Making of the flowcharts.
Ameerul Syahmi Bin Azamuddin	2018039	Scenario testing and screenshots.

Program Code

```cpp
#include <iostream>
#include <string>
#include <cstdlib>
#include <limits>

using namespace std;

//Global variable
int Total;

class Question
{
   private:
      string Question_Text;
      string Answer_1;
      string Answer_2;
      string Answer_3;
      string Answer_4;

      int Correct_Answer;
      int Question_Score;

   public:
      void setValues (string, string, string, string, string, int, int); //
      void askQuestion ();
};
// Uses 'Question' class - To take private variables from class
void Question::setValues (string q, string a1, string a2, string a3, string a4, int ca, int pa)
{
   Question_Text = q;
   Answer_1 = a1;
   Answer_2 = a2;
   Answer_3 = a3;
   Answer_4 = a4;
   Correct_Answer = ca;
   Question_Score = pa;
}
//Usee the 'Question' class - To display question, answer, and notes on whether it's correct or not
void Question::askQuestion()
{
      int Guess;
```

```cpp
        do{
    cout << endl;
    cout << Question_Text << endl; //print parameters given in q(i) setValues object.
    cout << "1. " << Answer_1 << endl;
    cout << "2. " << Answer_2 << endl;
    cout << "3. " << Answer_3 << endl;
    cout << "4. " << Answer_4 << endl;
    cout << endl;
    cout << "What is your answer?(using corresponding number)" << endl;
    cin >> Guess;
    cin.ignore();


        if ( Guess!=1 && Guess!=2 && Guess!=3 && Guess!=4 )
        {
                cout << "Error input, please answer using corresponding number, eg. 1,2,3,4" <<
endl;
                //Get only integer
                cin.clear();
        cin.ignore(numeric_limits<streamsize>::max(), '\n');
        }

        }while ( Guess!=1 && Guess!=2 && Guess!=3 && Guess!=4 );

    // if else to print if answer is correct or not
    if (Guess == Correct_Answer)
    {
       cout << endl;
       cout << "Great! You are correct" << endl;
       Total = Total + Question_Score;
       cout << endl;
    }
    else
    {
       cout << endl;
       cout << "Oh no! You are wrong." << endl;
       cout << "The correct answer is " << Correct_Answer << "." << endl;
       cout << endl;
    }
}

//
```

```cpp
//Function prototype
char Menu(); //Display Menu for user to enter choice, returning choice to main
char Notes(); //Display Notes, return choice if program wants to repeat
char Problem(); //Display Problem, return choice if program wants to repeat

int main()
{
        char choice; // "choice" is to tell program which function to call
        bool backtostart=false;        //"back to start" is to tell program to go back from the start
of main or return 0

do{ // do while loop if user wants to go back to main menu
choice=Menu();

        switch (choice) //Decide what function to call based on user input at Menu()
        {
                case 'a' :
                        if(Notes()=='Y'){
                        backtostart=true;
                        break;
                        }
                        else{
                        return 0;}

                        break;

                case 'b' :
                        if(Problem()=='Y'){
                        backtostart=true;
                        break;
                        }
                        else{
                        return 0;}

                        break;

                case 'c' : return 0;
        }
}while(backtostart==true);
        return 0;
}

char Menu()
{
```

```cpp
    char option='a';

    //Print menu, option for users do while loop for error if option is invalid
    do{
            if(option!='a' && option!='b' && option!='c')
            {
                    cout<<"Error, character is unavailable in option\n"<<endl;
            }

            cout<<">>--Learning Aid Software--<< \n"<<endl
                    <<"Newton's Laws of Motion \n"<<endl
                    <<" - Option Menu: - \n"<<endl
                    <<"a) Notes\n\nb) Quiz mcq\n\nc) Exit Program \n"<<endl
                    <<"*(Enter the character/letter to pick your option)"<<endl
                    <<"**(For Y/N prompts, please use their capital form)\n"<<endl;

            cin>>option;
        }while(option!='a' && option!='b' && option!='c');

    cout<<"\n\n";
    return option;
    }

    char Notes()
    {
            char option='Y'; //character to return

            cout<<"--Newton's First Law--"<<endl
                    <<"The first law states that as an object at rest will stay at rest, and an object in
    motion will stay in motion unless acted on by a net external force.\n"
                    <<"This law explains the inertia phenomena. eg, a person standing on a bus will
    continue to move forward even if the bus has stopped.\n\n"
                    <<"Mathematically:\n"
                    <<"If Fnet = 0, then a = 0"<<endl;

                    cout << "Press Enter to continue .. " << endl;
                    cin.get();
                    cin.get();

            cout<<"--Newton's Second Law--"<<endl
                    <<"Change of momentum over time is directly proportional to the force applied,
    for both are in the same direction."<<endl
                    <<"or"<<endl
```

```cpp
            <<"Force is directly proportional to mass times acceleration, for a constant
mass\n"<<endl
            <<"Mathematically:\n"
            <<"F=ma, for a constant mass"<<endl
            <<"but generalized as, F=d/dt(p)"<<endl;

            cout << "Press Enter to continue .. " << endl;
            cin.get();

        cout<<"--Newton's Third Law--"<<endl
            <<"All forces between two objects exist in equal magnitude and opposite
direction."<<endl
            <<"eg. If object 'a" exerts a force on to 'b' to the right,then 'b' exerts a force on 'a'
to the left\n"<<endl
            <<"Mathematically:\n"
            <<"Fa=-Fb, for the negative sign is a convention to say that it acts in the opposite
direction."<<endl;

            cout << "Press Enter to continue .. " << endl;
            cin.get();


//Print do you wish to continue and loop for wrong input
do{
            if(option!='Y' && option!='N')
        {
            cout<<"Error, please enter Y for yes or N for no\n"<<endl;
        }
        cout<<"Do you wish to continue?"<<endl
            <<"(Enter Y for yes and N for no)"<<endl;
        cin>>option;
        cout<<"\n\n";
}while(option!='Y' && option!='N');

        return option;
}

char Problem(){
char option='Y'; // character to return
//Question code

cout << "----Newtonian Mechanics MCQ----" << endl
        << "This MCQ quiz consists of 10 question, and each question contains 10 marks" <<
endl
```

```cpp
            << "The passing grade would be 50 and above." << endl;

cout << "Press Enter to start the quiz .. " << endl;
cin.get();
cin.get();

string Name;

cout <<"What is your name?" << endl;
cin >> Name;
cout << endl;
char Respond;
do{
cout << "Are you ready to take the quiz " << Name << "? Y/N" << endl;
cin >> Respond;
if (Respond == 'Y')
{
   cout << endl;
   cout << "OK, Goodluck!" << endl;
}
else if (Respond == 'N')
{
   cout << "OK, Goodbye!\n\n" << endl;

do{
        cout<<"\nDo you wish to continue?"<<endl
                <<"(Enter Y for yes and N for no)"<<endl;
        cin>>option;
        if (option!='Y' && option!='N')
        {
                cout<<"Error, Enter Y for yes or N for no\n"<<endl;
                cout<<"\n\n";
        }
        return option;
  }while(option!='Y' && option!='N');
}
   //Print do you wish to continue and loop for wrong input
else if (Respond!='Y' && Respond!='N')
{
        cout<<"Error, Enter Y for yes or N for no\n"<<endl;
}
}while(Respond!='Y' && Respond!='N');

 //Creating object q(i) all in class 'Question';
```

```
Question q1; //Class 'Question' object q1,q2,q3,... etc.
Question q2;
Question q3;
Question q4;
Question q5;
Question q6;
Question q7;
Question q8;
Question q9;
Question q10;

//Creating question with function setValues(...) in class 'Question" take 5strings and 2integer for
a parameter
//Parameter is ("question_text","answer1","answer2","answer3","answer4", correct answer in
integer, points for each answer
q1.setValues ("Losing speed as you ride a bike uphill, demonstrates which Newton's law?",
      "Newton's First Law",
      "Newton's Second Law",
      "Newton's Third Law",
      "Insufficient information to determine",
      2, // The position of the correct answer
      10); // The answer value in points

q2.setValues ("An object with no net force acting remains at rest or in motion with constant
velocity, demonstrates which Newton's law?'",
      "Newton's First law",
      "Newton's Second law",
      "Newton's Third law",
      "Insufficient information to determine",
      1, // The position of the correct answer
      10); // The answer value in points

q3.setValues ("Pushing against a wall, the wall pushes back against you, demonstrates which
Newton's law?'",
      "Newton's First law",
      "Newton's Second law",
      "Newton's Third law",
      "Insufficient information to determine",
      3, // The position of the correct answer
      10); // The answer value in points

q4.setValues ("When you sit on a chair, the resultant force on you is?",
      "Zero",
      "Up",
```

```
            "Down",
            "Depending on your weight",
            1, // The position of the correct answer
            10); // The answer value in points

    q5.setValues ("In the absence of an external force, a moving object will?",
            "Stop immediately ",
            "Slowly slows down, and then stop",
            "Go faster",
            "Move with a constant speed",
            4, // The position of the correct answer
            10); // The answer value in points

    q6.setValues ("You are standing on a moving bus, then you suddenly fall forward. You can imply
    that the bus's",
            "Velocity decreased",
            "Velocity increased",
            "Speed remained the same, but it's turning right",
            "Speed remained the same, but it's turning left",
            1, // The position of the correct answer
            10); // The answer value in points

    q7.setValues ("A constant net force acts on an object, what describes the motion of the object?",
            "constant acceleration",
            "constant speed",
            "constant velocity",
            "increasing acceleration",
            1, // The position of the correct answer
            10); // The answer value in points

    q8.setValues ("The acceleration of an object is inversely proportional to ",
            "the net force acting on it",
            "its position",
            "its velocity",
            "its mass",
            4, // The position of the correct answer
            10); // The answer value in points

    q9.setValues ("If you exert a force F on an object, the force which the object exerts on you will",
            "depend on whether or not the object is moving",
            "depend on whether or not you are moving",
            "depend on the relative masses of you and the object",
            "always be F",
            4, // The position of the correct answer
```

```
        10); // The answer value in points

q10.setValues ("A net force F accelerates a mass m with an acceleration a. If the same net
force is applied to mass 2m, then the acceleration will be",
        "4a",
        "2a",
        "a/2",
        "a/4",
        3, // The position of the correct answer
        10); // The answer value in points


 //Displaying q1 using function askQuestion while still having the question strings and answer
integer
q1.askQuestion();

q2.askQuestion();

q3.askQuestion();

q4.askQuestion();

q5.askQuestion();

q6.askQuestion();

q7.askQuestion();

q8.askQuestion();

q9.askQuestion();

q10.askQuestion();

cout << "Your Total Score is " << Total << " out of 100" << endl;

if (Total >= 50) // must be >=50 to pass the mcq quiz
{
    cout << "Great you passed the quiz!" << endl;
    cout << "Congratulations! Goodluck on your future studies." << endl;
Total=0;
}

else {
```

```cpp
cout << "Oh no! You failed the quiz." << endl;
cout << "Better luck next time" << endl;
Total=0;
}

//Print do you wish to continue and loop for wrong input
do{
                if(option!='Y' && option!='N')
        {
                cout<<"Error, please enter Y for yes or N for no\n"<<endl;
        }
        cout<<"\nDo you wish to continue?"<<endl
                <<"(Enter Y for yes and N for no)"<<endl;
        cin>>option;
        cout<<"\n\n";
}while(option!='Y' && option!='N');

        return option;
}
```

Abstract

This Learning Aid Software is built specifically for students of the Centre for Foundation Studies (CFS) in IIUM to grasp some physics concepts, mainly the **three laws of motion**. These laws of motion were developed by Sir Isaac Newton in 1686.

Briefly, the first law that Sir Isaac Newton has presented states that *an object at rest will stay at rest, and an object in motion will stay in motion unless acted on by a net external force*, this is often referred to as the **law of inertia**. Newton's second law states that *the rate of change in momentum of a body over a change in time is proportional to the force applied*. Commonly known as **F = ma** for constant mass. Finally, the third law states that *all forces between two objects exist in the same magnitude but opposite direction*. **If you slap a car, the car "slaps" back at you**. The three laws presented by Newton are the so-called fundamental laws in an area of studies called Classical Mechanics. These laws were verified experimentally for over 200 years, this is what makes it so important. The preferred prerequisites upon learning Newton's laws of motion are basic algebra, precalculus, and unit analysis. Nonetheless, Newtonian Mechanics are commonly known as the first or second subtopic in physics that you will learn throughout secondary school, and even college or university.

Hopefully, this learning aid software will benefit all the physical sciences and engineering students of CFS in IIUM. It will provide notes on Newton's laws and conceptual questions on the subject. But most importantly, it is really user friendly and does not require any fancy personal computer or smartphones to run. It does not even require the internet while running. Thus saving much more battery life, and therefore increasing the time for studying and not charging. Bugs do not exist in our software, thanks to our team in debugging. If it does, however, it is not significant.

Problem

Most Physics and Engineering students suffer in the long run when not trying to understand this subject enough. Newtonian Mechanics are by far the most intuitive concept in physics and engineering. So, learning this thoroughly can be like an approach for practicing how to grasp more non-intuitive subjects such as electricity and magnetism, etc.

Function's Flowchart and Description

In main():

int main()
- Main is the main function which acts as a leader. It determines on which function to call and whether to exit the program or start the program all over again.   - First it calls *Menu()*, then *Menu()* returns a value choice. *main()* determines which function to call next. If choice is 'a', main calls *Notes()*. If choice is 'b' , main calls *Problem()*. If the choice is 'c'. It exits the program.   - For *Notes()* and *Problem()* is called, they return a value *backtostart*. Main determines on whether to redo the program or exit. This is by checking if the value *backtostart*.

In class Question: In Question::setValues(5strings 2int):

```
                Question
─────────────────────────────────
- Question_Text : string
- Answer_1 : string
- Answer_2 : string
- Answer_3 : string
- Answer_4 : string
- Correct_Answer : int
- Question_Score : int
─────────────────────────────────
+ setValues ( : string, string,
            string, string,
            string, int, int) :
            void

+ askQuestion() : void
```

Question::setValues(string q, string a1, string
                    a2, string a3, string a4,
                    int ca, int pa)

```
Question_Text = q;
Answer_1 = a1;
Answer_2 = a2;
Answer_3 = a3;
Answer_4 = a4;
Correct_Answer = ca;
Question_Score = pa;
```

End

class Question	setValues(5string, 2int)
● This class creates/stores 7 private variables for any of the *Question* members to access. ● Also, creates 2 functions called: ● *setValues* that take 7 parameters, that is 5 strings and 2 integers. *setValues* does not return any value. ● *askQuestion()* takes no parameters and returns no value.	● This function takes all the private variables from the class for functions *Problem()* to use, since it is not a member of the class *Question*.

Question::askQuestion()

```
         ┌─────────────────────────┐
         │  Question::askQuestion()│
         └────────────┬────────────┘
                      │
         ┌────────────▼────────────┐
         │  display Question_Text  │◄──────────────────────┐
         └────────────┬────────────┘                       │
                      │                                     │
         ┌────────────▼────────────┐                        │
         │ display Answer_1 ,2,3.. │                        │
         │   until Answer_4        │                        │
         └────────────┬────────────┘                        │
                      │                                     │
         ┌────────────▼────────────┐                        │
         │ ask user to input number│                        │
         │ corresponds to which    │                        │
         │ answer that user think  │                        │
         │ is correct              │                        │
         └────────────┬────────────┘                        │
                      │                                     │
              ┌───────▼───────┐                   ┌─────────┴────────┐
              │ Guess not     │───Yes───────────► │ display Error    │
              │ 1,2,3, and 4? │                   │ input            │
              └───────┬───────┘                   └──────────────────┘
                    No│
              ┌───────▼───────┐         ┌────────────────────────────┐
              │ Guess ==      │──No────►│ display "wrong, the correct │
              │ Correct_Answer?│        │ answer is Correct_Answer"  │
              └───────┬───────┘         └──────────────┬─────────────┘
                  Yes │                                │
         ┌────────────▼────────────┐          ┌────────▼────────┐
         │ display "You are        │          │      End        │
         │ correct"                │          └────────▲────────┘
         └────────────┬────────────┘                   │
                      │                                 │
         ┌────────────▼────────────┐                    │
         │ Total+=Question_Score   │────────────────────┘
         └─────────────────────────┘
```
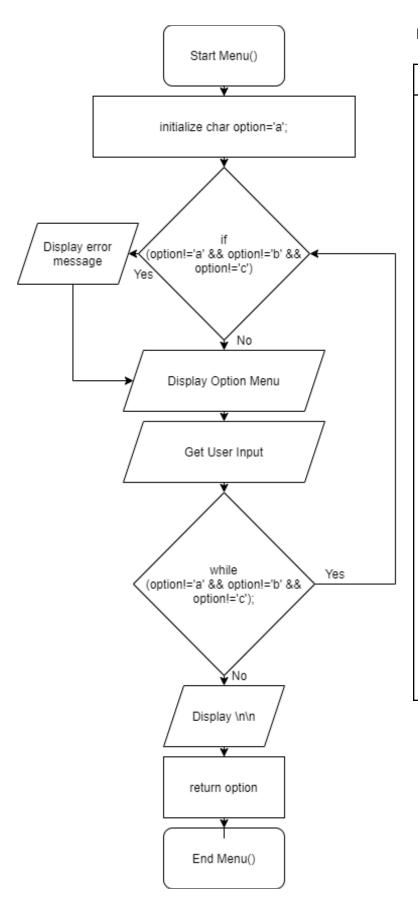
InQuestion::askQuestion():

askQuestion()

- This function gets called in *Problem()* to display all the MCQ questions and answers.

- After that, it takes and checks the user's guesses on whether it is correct or not. Then display the right answer.

- Also, it Sums up all the marks for all 10 questions (1 correct answer = 10marks).

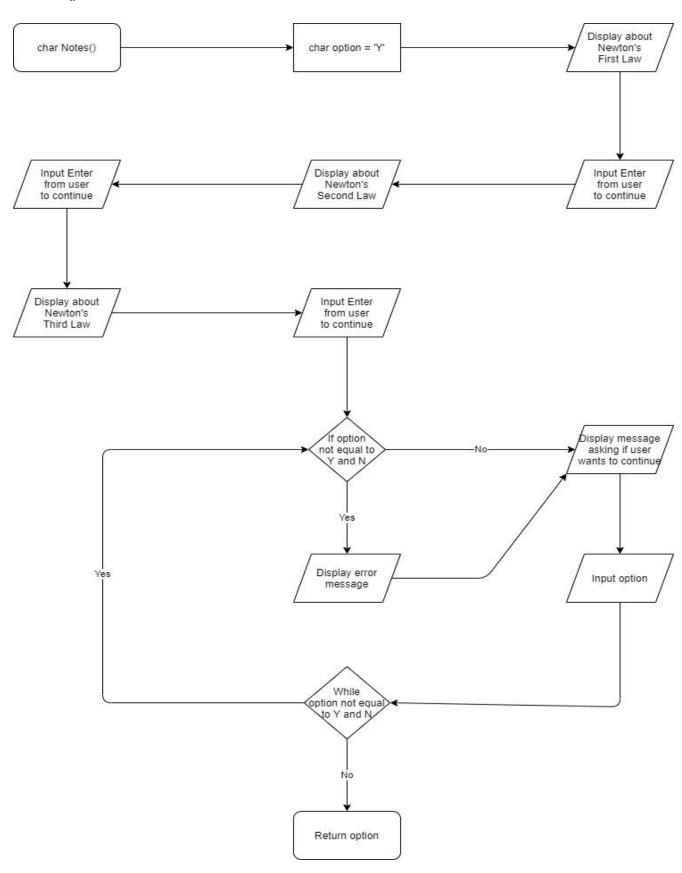- Last but not least, it doesn't return any value or take any parameter.

In Menu ():

```
Start Menu()
        |
        v
initialize char option='a';
        |
        v
      if
(option!='a' && option!='b' &&   --Yes--> Display error message
   option!='c')
        |
        | No
        v
Display Option Menu  <----
        |
        v
Get User Input
        |
        v
     while
(option!='a' && option!='b' &&   --Yes-->
   option!='c');
        |
        | No
        v
Display \n\n
        |
        v
return option
        |
        v
   End Menu()
```
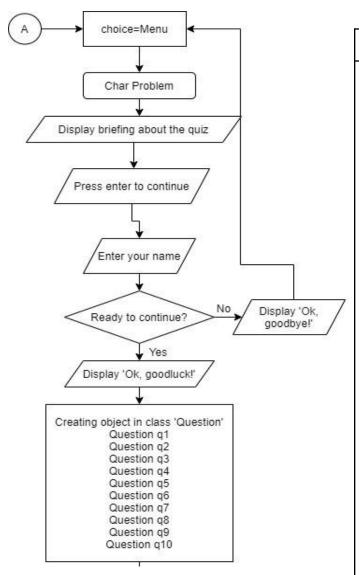
Menu()

- This function is to show a list of option menus for the user and get user choice when using the learning aid program.

- First, the function initializes the "char option" variable with the character 'a'.

- Next, the function goes into a "do-while" loop and runs through an "if" command to check whether the value satisfies the conditions.

- If yes, the function displays an error message and displays/re-displays the option menu.

- If no, then the function displays the option menu and gets the user input.

- After getting user input, the function postchecks the value of "char option".

- If the condition is satisfied, the function loops back to the "if" command and displays an error message before re-displaying the option menu.

- If the condition is not satisfied, the function displays 2 newlines before returning the "char option" value to "char choice" in "main ()".

In Notes():

```
char Notes()  →  char option = 'Y'  →  Display about Newton's First Law
```

```
Input Enter from user to continue  ←  Display about Newton's Second Law  ←  Input Enter from user to continue
```

```
Display about Newton's Third Law  →  Input Enter from user to continue
```

If option not equal to Y and N

— No → Display message asking if user wants to continue

Yes → Display error message

Display error message → Display message asking if user wants to continue

Display message asking if user wants to continue → Input option

Input option → While option not equal to Y and N

While option not equal to Y and N — Yes → (back up to If option not equal to Y and N)

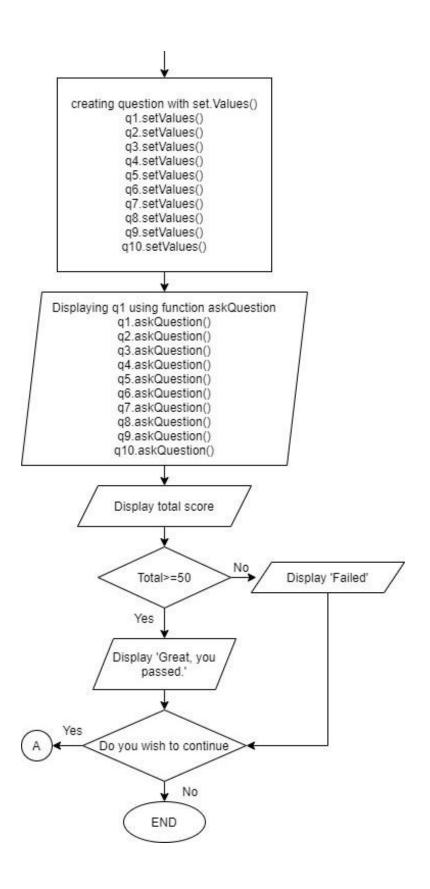While option not equal to Y and N — No → Return option

Notes()
● This function gets called by main() to display the notes about three Newton's laws. ● It will display Newton's law one by one, it will pause to let the program be more interactive and less boring. Users can enter input to let the program display the next Newton's law. ● After that, it checks if the option is not Y and N, and it will display an error message if it is correct. ● After it checks, the program will display a message to ask if the user wants to continue. And the user can input the option. ● Lastly, it will check if the option is not Y and N, if it is not Y and N, the program will loop the messages as in the flowchart.

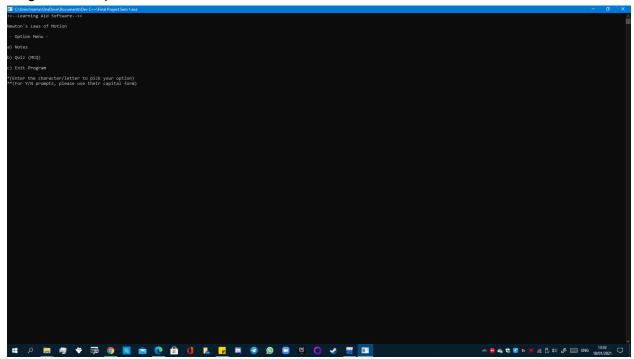## Flowchart

```
 ( A ) ──────►  choice=Menu  ◄──────┐
                    │               │
                    ▼               │
              Char Problem          │
                    │               │
                    ▼               │
     / Display briefing about the quiz /
                    │               │
                    ▼               │
       / Press enter to continue /  │
                    │               │
                    ▼               │
          / Enter your name /       │
                    │               │
                    ▼               │
          ◇ Ready to continue? ──No──►  / Display 'Ok, goodbye!' /
                    │ Yes
                    ▼
       / Display 'Ok, goodluck!' /
                    │
                    ▼
   ┌─────────────────────────────────┐
   │ Creating object in class 'Question' │
   │           Question q1           │
   │           Question q2           │
   │           Question q3           │
   │           Question q4           │
   │           Question q5           │
   │           Question q6           │
   │           Question q7           │
   │           Question q8           │
   │           Question q9           │
   │           Question q10          │
   └─────────────────────────────────┘
                    │
```

In Problem():

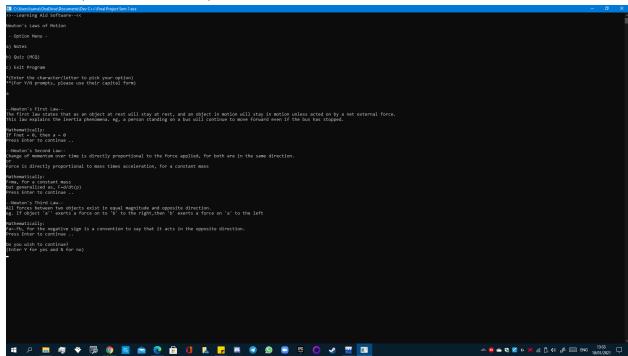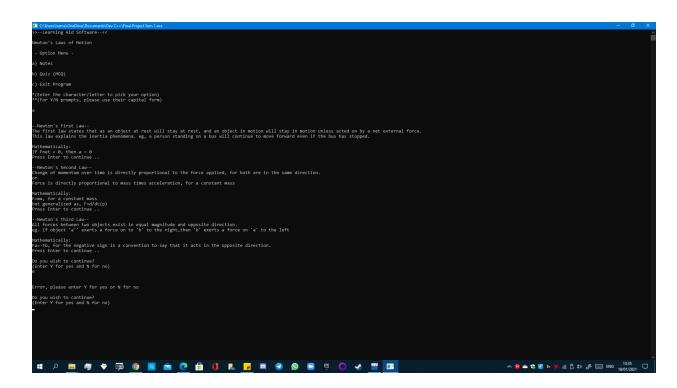| Problem() |
| --- |
| ● This function gets called by main() to display the notes about Menu. |
| ● It will ask the user to enter his/her name. |
| ● If he/she is ready to answer, it displays 'Ok, goodluck', if not, it will display 'Ok, goodbye'. |
| ● Then, we will create an object for the class 'Question'. For example 'Question q1'. |
| ● Then, we created questions with function setValues(...) in class 'Question" take 5strings and 2integer for a parameter. |
| ● The parameters are ("question_text", "answer1", "answer2", "answer3", "answer4", correct answer in integer, points for each answer). |
| ● After that, it will display q1 (question) using function askQuestion while still having the question strings and answer integer. |
| ● After answering all the questions, it will display total marks and they got marks above 50, it will display "Great, you passed", if below 50, it will display "Failed". |
| ● Lastly, it asks the user if they wish to continue or not. If yes, it will go back to the Menu function, if not, the function will end. |

```
creating question with set.Values()
        q1.setValues()
        q2.setValues()
        q3.setValues()
        q4.setValues()
        q5.setValues()
        q6.setValues()
        q7.setValues()
        q8.setValues()
        q9.setValues()
        q10.setValues()
```

```
Displaying q1 using function askQuestion
        q1.askQuestion()
        q2.askQuestion()
        q3.askQuestion()
        q4.askQuestion()
        q5.askQuestion()
        q6.askQuestion()
        q7.askQuestion()
        q8.askQuestion()
        q9.askQuestion()
        q10.askQuestion()
```

Display total score

Total>=50 — No → Display 'Failed'

Yes

Display 'Great, you passed.'

Yes → A

Do you wish to continue

No

END

Scenario Tests and Screenshots

Program Startup:

Scenario A (User selects Notes):

- If user enters invalid input (next page)

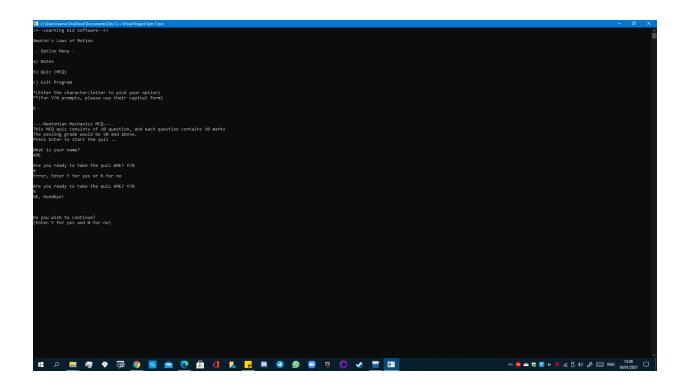**-** If user wants to continue using the program

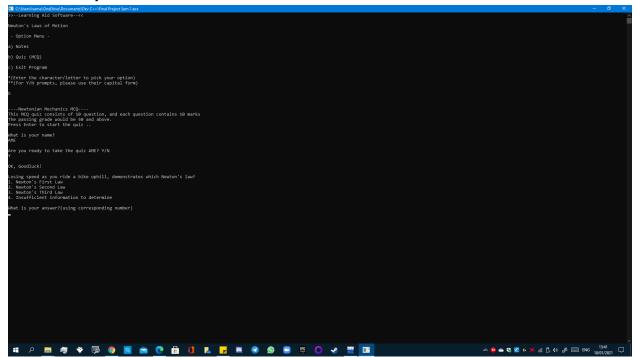**-** If user does not want to continue using the program (next page)

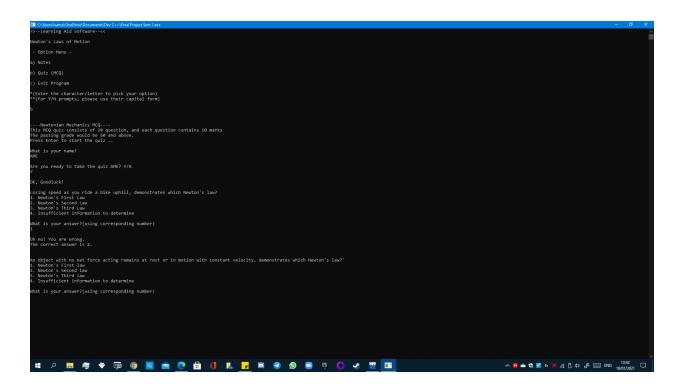Scenario B (User selects Quiz (MCQ)):

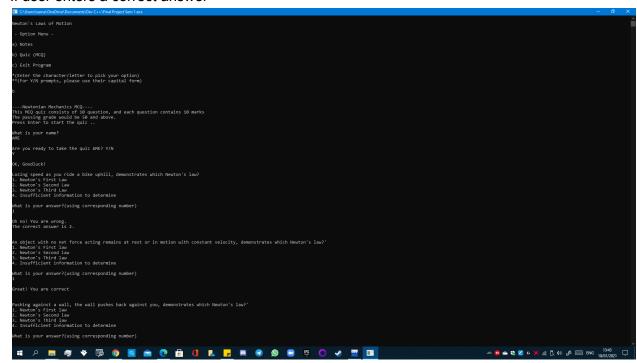- If user enters invalid input/is not ready (next page)
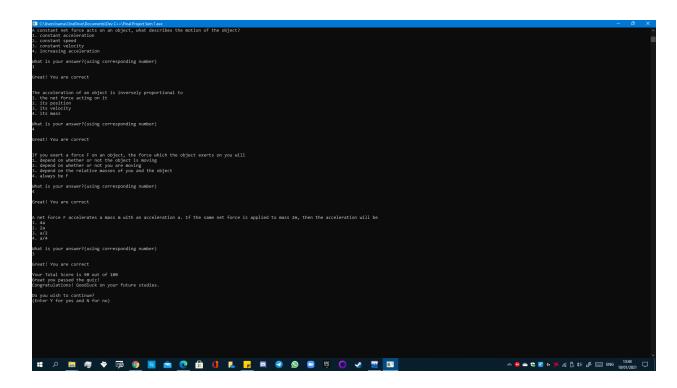
- If user is ready

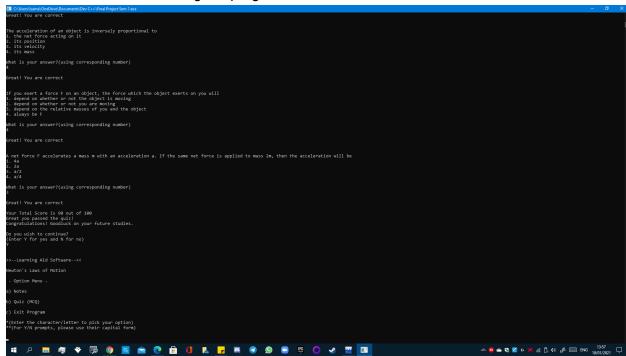- If user enters a wrong answer (next page)

- If user enters a correct answer

- Total marks display (next page)
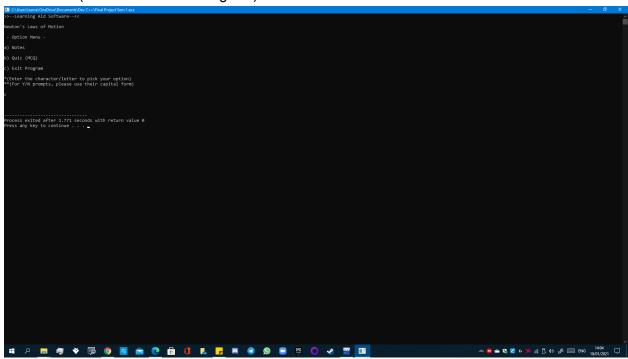
- If user wants to continue using the program

- If user enters an invalid input/does not want to continue using the program

Scenario C (User selects Exit Program):

Scenario D (User enters invalid input in the Menu) (next page):

b) Quiz (MCQ)

c) Exit Program

*(Enter the character/letter to pick your option)
**(For Y/N prompts, please use their capital form)

A
Error, character is unavailable in option

>>--Learning Aid Software--<<

Newton's Laws of Motion

 - Option Menu -

a) Notes

b) Quiz (MCQ)

c) Exit Program

*(Enter the character/letter to pick your option)
**(For Y/N prompts, please use their capital form)

z
Error, character is unavailable in option

>>--Learning Aid Software--<<

Newton's Laws of Motion

 - Option Menu -

a) Notes

b) Quiz (MCQ)

c) Exit Program

*(Enter the character/letter to pick your option)
**(For Y/N prompts, please use their capital form)