1. **Make a program that calculate and display the Body Mass Index (BMI) of a person. The program is should read the height and weight of a person being given and display the Body Mass Index (BMI).**

Use to calculate the Body Mass Index (BMI) $= \frac{\textbf{weight}}{\textbf{height} * \textbf{height}}$

➢ Problem analysis

Input

- ✓ Weight of a person
- ✓ Height of a person

Output

- ✓ Display BMI of a person

➢ Process or Operation

- Variable declaration: that holds an input and operational results.
- Print input prompt message and read corresponding input.
- Calculate the BMI of a person.
- Print output prompt message and process result (BMI).

➢ Design the program

- Variable declaration (What and How)

Float weight, height, BMI;

- Reading input data (How)

Cout << "please enter the weight and the height of the person ";

Cin >> weight, >> height;

$$\textbf{BMI} = \frac{\textbf{weight}}{\textbf{height} * \textbf{height}}$$

Cout << "The BMI of the person is: " << BMI << "Kg/m^2 ";

2. **Write a program that prompts should read the capacity in gallons of an automobile fuel tank and the miles per gallons the automobiles can be driven. the program out puts the number of miles the automobile can be driven without refueling.**

➢ Problem analysis

Input

- ✓ Tank capacity
- ✓ Mile_per_gallon

Output

- ✓ Display num_mile

➢ Process or Operation

- Variable declaration: that holds an input and operational results.
- Initializing mile_per_gallons to zero.
- Print input prompt message and read corresponding input data.
- Calculate the miles or number of miles.
- Print output prompt message and process result (num_mile).

➢ Design the program

- Variable declaration and Initialization (What and How).

Float tankCapacity, milePerGallon, numMiles;

- Reading input data (How)

Cout << "please enter the capacity in gallons of fuel and the miles per gallons ";

Cin >> tankCapacity, >> milePerGallon;

numMile = tankCapacity * milePerGallon;

Cout << "The miles of the automobile can be driven without refueling is: " << BMI << "Kg/m^2";

3. **Write program that converts a letter entered from the keyboard to its uppercase or lowercase equivalent (Hint: use the function defined in ctype.h library header file).**

➢ Problem analysis

Input

✓ Letter

✓

Output

✓ Uppercase letter

✓ Lowercase letter

➢ Process or Operation

- Variable declaration: that holds an input from the keyboard and operational results.
- Print input prompt message and read corresponding input data.
- Converts a letter entered from the keyboard.
- Print output prompt message and process result (uppercase and lowercase letter).

➢ Design the program

- Variable declaration (What and How)

```
char letter, uppercaseLetter, lowercaseLetter;
```

- Reading input data (How)

```
Cout << "please enter the letter by using your keyboard ";

Cin >> letter;

convert to uppercase or lowercase

if (isalpha(letter)) { chr upercaseLetter = toupper(letter)

                    chr lowercaseLetter = tolower(letter)

                    cout << "upper letter" <<uppercaseLetter<< "lower letter" <<lowercaseLetter;}

else {cout << "Invalid,….."}
```

4. **Write a program that find the result of the expression x^y where the value of x and y are entered by the user.**

➢ Problem analysis

Input

- ✓ X value
- ✓ Y value

Output

- ✓ Display result of expression

➢ Process or Operation

- Variable declaration: that holds an input and operational results.
- Print input prompt message and read corresponding input data.
- Calculate the result of the expressions $\mathbf{(x^Y)}$.
- Print output prompt message and process result $\mathbf{(x^y)}$.

➢ Design the program

- Variable declaration (What and How).

```
Float x, y, result;
```

- Reading input data (How)

```
Cout << "please enter the x value and the y value ";
Cin >> x, >> y;
```

Result = pow(x, y);

```
Cout << "The result of the expression is: " << x << "^" <<y << " = " <<result;
```

5. **Design an algorithm that write a to read an employee name weekly working hours, bonus rate per hour and base salary and find the employee gross salary. net salary and bonus payment. (Hint: pension rate = 5%, tax= 15%)**

➢ **Problem analysis**

Input

✓ Employee's name, Bonus_rate,

✓ Weekly_hour, Base_salary

✓ hrs_worked

Output

✓ Display gross_salary

✓ Net_salary, Bonus_payment

➢ **Process or Operation**

- Variable declaration: that holds an input and operational results.
- Defining the pension and tax rate as constant.
- Print input prompt message and read corresponding input data.
- Calculate the Gross salary, bonus payment, the deduction for pension and tax rate, and Net salary.
- Print output prompt message and process result**.**

➢ **Design the program**

- **Variable declaration (What and How).**

```
string employeeName;    double weeklyHours, baseSalary;
double overtimeHour, bonusPayment, bonusRate;              int hrsWorked;
```

- **Reading input data (How)**

```
cout << "Enter the employee's name: ";        cin >>employeeName;
if (hrsWorked > 40){ cout <<"Enter worked hours: ";      cin >>hrsWorked; ……}
double grossSalary = baseSalary + bonusPayment;   double pension = 0.05 * grossSalary;
double taxRate = 0.15 * grossSalary;  double netSalary = grossSalary - (pension + taxRate);
cout << "Gross Salary: " <<grossSalary <<endl;
```

6. **A serial transmission line can transmit 960 characters a second. write a program that will calculate how long it will take to send a file, given the file size, test your program on a 400MB (419430400byte) file which may take days.**

➢ Problem analysis

Input

✓ File_size

✓

Output

✓ Time_taken

➢ Process or Operation

- Variable declaration to store files: that holds an input from keyboard and operational results.
- Print input prompt message and read corresponding input data.
- Calculate transmission time(time taken).
- Print output prompt message and process result Time taken.

➢ Design the program

- Variable declaration to store the file size. (What and How)

```
const int transmission_speed = 960;     int file_size;
```

- Reading input data (How)

```
cout << "please enter the size of file to be sent in bytes: ";    cin >> file_size;

calculate transmission time

double time_taken = (file_size)/(transmission_speed);

cout << "Estimated time to send the file: " << time_taken << " seconds" <<endl;
```