```c
/*
1. WAP to declare and initialize all data types e.g. int, float, char, double, long, short, long double, long long and print their size (in Bytes) using the “sizeof” operator. [ “%lu”]
*/
#include <stdio.h>

int main() {
   // Declare and initialize variables
   int intType = 42; // An integer
   float floatType = 3.14; // A floating-point number
   double doubleType = 2.71828; // A double-precision floating-point number
   char charType = 'A'; // A character
   long long longLongType = 1234567890; // A long long integer
   short shortType = 32767; // A short integer
   long double longDoubleType = 1.234567890123456; // A long double

   // Print sizes of each data type
   printf("Size of int: %zu bytes\n", sizeof(intType));
   printf("Size of float: %zu bytes\n", sizeof(floatType));
   printf("Size of double: %zu bytes\n", sizeof(doubleType));
   printf("Size of char: %zu byte\n", sizeof(charType));
   printf("Size of long long: %zu bytes\n", sizeof(longLongType));
   printf("Size of short: %zu bytes\n", sizeof(shortType));
   printf("Size of long double: %zu bytes\n", sizeof(longDoubleType));

   return 0;
}
```

Output :

```
Size of int: 4 bytes
Size of float: 4 bytes
Size of double: 8 bytes
Size of char: 1 byte
Size of long long: 8 bytes
Size of short: 2 bytes
Size of long double: 12 bytes

Process returned 0 (0x0)   execution time : 0.569 s
Press any key to continue.
```

```
/*
2. WAP to print your name, roll number, branch, marks which are stored in appropriate data types. [ char name[]=”xyzw”]
*/

#include <stdio.h>
#include <string.h> // Include this header for string functions

// Define the student structure
struct Student {
   char name[50]; // Store student name (up to 50 characters)
   char rollNumber[10]; // Store roll number (up to 10 characters)
   char branch[10]; // Store branch (up to 10 characters)
   float marks; // Store marks (floating-point value)
};

int main() {
   struct Student s; // Create a structure variable for a student

   // Initialize student information
   strcpy(s.name, "Prashant Ranjan"); // Set student name
   strcpy(s.rollNumber, "24MC3035"); // Set roll number
   strcpy(s.branch, "MATHEMATICS AND COMPUTING "); // Set branch
   s.marks = 100.0; // Set marks

   // Display student information
   printf("Name: %s\n", s.name); // Print student name
   printf("Roll Number: %s\n", s.rollNumber); // Print roll number
   printf("Branch: %s\n", s.branch); // Print branch
   printf("Marks: %.2f\n", s.marks); // Print marks with 2 decimal places

   return 0; // Exit program
}
```

Output :

```
Name: Prashant Ranjan
Roll Number: 24MC3035
Branch: MATHEMATICS
Marks: 100.00

Process returned 0 (0x0)   execution time : 0.673 s
Press any key to continue.
```

```c
/*
3. WAP to find Fahrenheit of a Celsius. formula:- “C*9/5+32”
*/

#include <stdio.h>

int main() {
   float celsius, fahrenheit;

   // Input temperature in Celsius
   printf("Enter temperature in Celsius: ");
   scanf("%f", &celsius);

   // Convert to Fahrenheit using the formula: Fahrenheit = Celsius * 9/5 + 32
   fahrenheit = celsius * 9/5 + 32;

   // Display temperature in Fahrenheit
   printf("Temperature in Fahrenheit: %.2f\n", fahrenheit);

   return 0;
}
```

Output :

```
Enter temperature in Celsius: 10
Temperature in Fahrenheit: 50.00

Process returned 0 (0x0)   execution time : 3.580 s
Press any key to continue.
```

```
/*
4. WAP to find Fahrenheit of a Celsius. formula:- “C*(9/5)+32”
*/
#include <stdio.h>

int main() {
   float celsius, fahrenheit;

   // Input temperature in Celsius
   printf("Enter temperature in Celsius: ");
   scanf("%f", &celsius);

   // Convert to Fahrenheit using the formula: Fahrenheit = (Celsius * 9/5) + 32
   fahrenheit = (celsius * 9/5) + 32;

   // Display temperature in Fahrenheit
   printf("Temperature in Fahrenheit: %.2f\n", fahrenheit);

   return 0;
}
```

Output :

```
Enter temperature in Celsius: 25
Temperature in Fahrenheit: 77.00

Process returned 0 (0x0)   execution time : 5.672 s
Press any key to continue.
```

```
/*
5 (i). WAP to find the value of addition of two numbers (i) int
*/
#include <stdio.h>

int main() {
   int number1, number2, sum; // Declare variables for input and result

   // Input two integers
   printf("Enter two integers: ");
   scanf("%d %d", &number1, &number2);

   // Calculate the sum
   sum = number1 + number2;

   // Display the result
   printf("%d + %d = %d\n", number1, number2, sum);

   return 0;
}
```

Output :

```
Enter two integers: 55
25
55 + 25 = 80

Process returned 0 (0x0)   execution time : 8.417 s
Press any key to continue.
```

```
/*
5(ii). WAP to find the value of addition of two numbers (ii) double.
*/

#include <stdio.h>

int main() {
   double number1, number2, sum; // Declare variables for input and result

   // Input two numbers
   printf("Enter two numbers: ");
   scanf("%lf %lf", &number1, &number2);

   // Calculate the sum
   sum = number1 + number2;

   // Display the result
   printf("Sum: %.2lf\n", sum);

   return 0;
}
```

Output :

```
Enter two numbers: 25.054
75.109
Sum: 100.16

Process returned 0 (0x0)   execution time : 15.566 s
Press any key to continue.
```

```
/*
6. WAP to find the division of two two numbers (i) int (ii) double (iii) int/double (iv)
double/int.
*/

#include <stdio.h>

int main() {
   int num1, num2; // Declare variables for integer division
   double num3, num4; // Declare variables for double division

   // Input two integers
   printf("Enter two integers: ");
   scanf("%d %d", &num1, &num2);

   // Calculate integer division
   int intResult = num1 / num2;

   // Input two doubles
   printf("Enter two decimal numbers: ");
   scanf("%lf %lf", &num3, &num4);

   // Calculate double division
   double doubleResult = num3 / num4;

   // Display results
   printf("Integer division result: %d\n", intResult);
   printf("Double division result: %.2lf\n", doubleResult);

   // Calculate mixed division (double/int)
   double mixedResult = num3 / num2;
   printf("Mixed division (double/int) result: %.2lf\n", mixedResult);

   // Calculate mixed division (int/double)
   mixedResult = num1 / num4;
   printf("Mixed division (int/double) result: %.2lf\n", mixedResult);

   return 0;
}
```

Output :

```
Enter two integers: 200
15
Enter two decimal numbers: 209.10
27.95
Integer division result: 13
Double division result: 7.48
Mixed division (double/int) result: 13.94
Mixed division (int/double) result: 7.16

Process returned 0 (0x0)   execution time : 28.438 s
Press any key to continue.
```

```c
/*
7. WAP to take an integer array of 5 elements and print its value at indices (i) 0, (ii)3, (iii)4
*/
#include <stdio.h>

int main() {
   int myArray[5]; // Declare an integer array of size 5

   // Input 5 integers from the user and store them in the array
   printf("Enter 5 integers:\n");
   for (int i = 0; i < 5; ++i) {
      scanf("%d", &myArray[i]);
   }

   // Print the values at specific indices
   printf("Value at index 0: %d\n", myArray[0]);
   printf("Value at index 3: %d\n", myArray[3]);
   printf("Value at index 4: %d\n", myArray[4]);

   return 0;
}
```

Output :

```
Enter 5 integers:
5
15
25
30
55
Value at index 0: 5
Value at index 3: 30
Value at index 4: 55

Process returned 0 (0x0)   execution time : 16.717 s
Press any key to continue.
```

```
/*
8. WAP to show BODMAS rule using integer and double.
*/
#include <stdio.h>

int main() {
   // Integer examples
   int intResult1 = 7 + (8 - 3 * 2); // Result: 9
   int intResult2 = 25 - 5 / (3 + 2); // Result: 20
   int intResult3 = 10 + 6 * (1 + 10); // Result: 76
   int intResult4 = 5 * (3 + 2) + 5; // Result: 30
   int intResult5 = 2 * (105 + 206) - 550 / 5; // Result: 412

   // Double examples
   double doubleResult1 = 1.0 / 3.0 * (6.0 + 8.0 * 3.0 - 2.0); // Result: 65.04
   double doubleResult2 = 18.0 / 10.0 - 4.0 + 32.0 / (4.0 + 10.0 / 2.0 - 1.0); // Result: 1.8

   // Print results
   printf("Integer results:\n");
   printf("1. %d\n2. %d\n3. %d\n4. %d\n5. %d\n", intResult1, intResult2, intResult3, intResult4,
intResult5);

   printf("\nDouble results:\n");
   printf("1. %.2lf\n2. %.2lf\n", doubleResult1, doubleResult2);

   return 0; // Exit program
}
```

Output :

```
Integer results:
1. 9
2. 24
3. 76
4. 30
5. 512

Double results:
1. 9.33
2. 1.80

Process returned 0 (0x0)   execution time : 0.755 s
Press any key to continue.
```

Email Id: 24MC3035@rgipt.ac.in
Branch: MATHEMATICS AND COMPUTING
Name: PRASHANT RANJAN
Roll No: 24MC3035