



Course Assignment Report

Only for course Teacher						
		Needs Improvement	Developing	Sufficient	Above Average	Total Mark
Allocate mark & Percentage		25%	50%	75%	100%	5
Clarity	1					
Content Quality	2					
Spelling & Grammar	1					
Organization and Formatting	1					
Total obtained mark						
Comments						

Semester: Fall-2023

Student Name: Hishamul Islam Towhid

Student ID: 0242220005341015

Batch: 39 Section: A

Course Code: SE133 Course Name: Software Development & Capstone project

Course Teacher Name: MD. Shohel Arman

Designation: Assistant Professor, Department of Software Engineering,DIU

Submission Date: 15/12/2023

1.Print all natural numbers from 1 to n.

```
#include <stdio.h>
```

```
int main() {
    int n;
    printf("Enter the value of n: ");
    scanf("%d", &n);

    for (int i = 1; i <= n; i++) {
        printf("%d ", i);
    }
}
```

```
return 0;
}
```

The screenshot shows a terminal window in a dark-themed IDE interface. The terminal output is as follows:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile }
if (?) { .tempCodeRunnerFile }
Enter the value of n: 2
1 2
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

2.Print sum of even numbers between 1 to n.

```

#include <stdio.h>

int main() {

    int n, sum = 0;

    printf("Enter the value of n: ");

    scanf("%d", &n);

    for (int i = 2; i <= n; i += 2) {

        sum += i;

    }

    printf("Sum of even numbers from 1 to %d: %d\n", n, sum);

    return 0;
}

```

The screenshot shows a code editor interface with a dark theme. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. The current file is 'problem1.c'. The code itself is a simple C program that prompts the user for a value of n, calculates the sum of even numbers from 1 to n, and prints the result. The terminal window below shows the command to change directory to 'oop\Capston', the compilation of the code with 'gcc', and the execution of the compiled binary. The output of the program is displayed as well.

```

PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc t
empCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter the value of n: 3
Sum of even numbers from 1 to 3: 2
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>

```

3.Print sum of even numbers in given range.

```
#include <stdio.h>
```

```
int main() {
    int start, end, sum = 0;
    printf("Enter the range (start end): ");
    scanf("%d %d", &start, &end);
```

```
    for (int i = start; i <= end; i++) {
        if (i % 2 == 0) {
            sum += i;
        }
    }
```

```
    printf("Sum of even numbers in the range %d to %d: %d\n", start, end, sum);
```

```
    return 0;
}
```

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar has icons for file types like Java (.java), JavaScript (.js), HTML (.html), and C/C++ (.c). The main editor area contains a C program named 'problem1.c' with the following code:

```
#include <stdio.h>
int main() {
    int start, end, sum = 0;
    printf("Enter the range (start end): ");
    scanf("%d %d", &start, &end);
    for (int i = start; i <= end; i++) {
        if (i % 2 == 0) {
            sum += i;
        }
    }
    printf("Sum of even numbers in the range %d to %d: %d\n", start, end, sum);
    return 0;
}
```

The terminal tab at the bottom shows the command line output:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter the range (start end):
```

The status bar at the bottom displays system information: 27°C Haze, Search, Cloud, Mail, Facebook, Google, YouTube, and a date/time stamp: 9:06 PM 11/15/2023.

4. Print all odd numbers from 1 to n.

```
#include <stdio.h>
```

```
int main() {
```

```
    int n;
```

```
    printf("Enter the value of n: ");
```

```
    scanf("%d", &n);
```

```
    for (int i = 1; i <= n; i += 2) {
```

```
        printf("%d ", i);
```

```
}
```

```
return 0;
```

```
}
```

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the value of n: ");
    scanf("%d", &n);
    for (int i = 1; i <= n; i += 2) {
        printf("%d ", i);
    }
    return 0;
}
```

PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if (\$?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (\$?) { ./tempCodeRunnerFile }

Enter the value of n: 3

1 3

5. Print odd numbers in given range.

```
#include <stdio.h>
```

```
int main() {

    int start, end;

    printf("Enter the range (start end): ");

    scanf("%d %d", &start, &end);
```

```
for (int i = start; i <= end; i++) {
```

```
    if (i % 2 != 0) {
```

```
        printf("%d ", i);
```

```
}
```

```
}
```

```
return 0;
```

```
}
```

A screenshot of the Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar shows icons for various files: Calculation.java, calT.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (the active file), Car.java, and Student.java. The main editor area contains the following C code:

```
1 #include <stdio.h>
2
3 int main() {
4     int start, end;
5     printf("Enter the range (start end): ");
6     scanf("%d %d", &start, &end);
7
8     for (int i = start; i <= end; i++) {
9         if (i % 2 != 0) {
10             printf("%d ", i);
11         }
12     }
13
14     return 0;
15 }
16
```

The terminal below the editor shows the command line output:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc t empCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter the range (start end): 2 3
3
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The bottom status bar displays file information (Ln 14, Col 14, Spaces: 4, CRLF), connection details (Port: 5500, Win32), and system status (27°C Haze, 9:09 PM, 11/15/2023).

6. Print all factors of a number.

```
#include <stdio.h>
```

```
int main() {
```

```
    int num;
```

```
    printf("Enter a number: ");
```

```
    scanf("%d", &num);
```

```
    printf("Factors of %d are: ", num);
```

```
    for (int i = 1; i <= num; i++) {
```

```
        if (num % i == 0) {
```

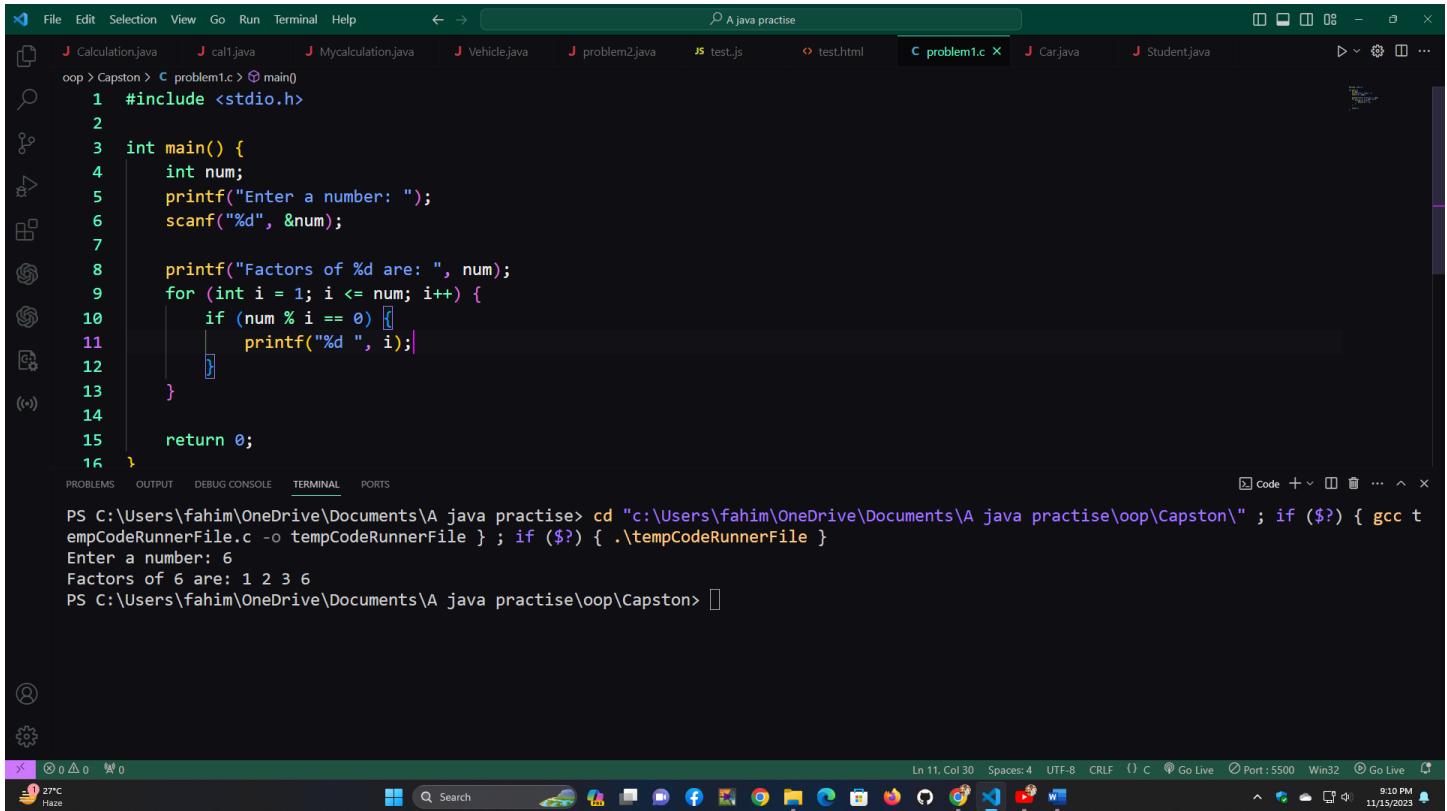
```
            printf("%d ", i);
```

```
}
```

```
}
```

```
    return 0;
```

```
}
```



```
#include <stdio.h>
int main() {
    int num;
    printf("Enter a number: ");
    scanf("%d", &num);
    printf("Factors of %d are: ", num);
    for (int i = 1; i <= num; i++) {
        if (num % i == 0) {
            printf("%d ", i);
        }
    }
    return 0;
}
```

PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if (\$?) { gcc tempCodeRunner.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }

Enter a number: 6

Factors of 6 are: 1 2 3 6

PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>

7. Print sum of odd numbers from 1 to n.

```
#include <stdio.h>
```

```
int main() {
    int n, sum = 0;
    printf("Enter the value of n: ");
    scanf("%d", &n);
```

```
    for (int i = 1; i <= n; i += 2) {
```

```
        sum += i;
```

```
}
```

```
    printf("Sum of odd numbers from 1 to %d: %d\n", n, sum);

    return 0;

}
```

A screenshot of the Visual Studio Code interface. The top bar shows tabs for various files: Calculation.java, cal1.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (the active tab), Car.java, and Student.java. The left sidebar has icons for file operations like Open, Save, Find, and Run. The main editor area contains the following Java code:

```
3 int main() {
4     int n, sum = 0;
5     printf("Enter the value of n: ");
6     scanf("%d", &n);
7
8     for (int i = 1; i <= n; i += 2) {
9         sum += i;
10    }
11
12    printf("Sum of odd numbers from 1 to %d: %d\n", n, sum);
13
14    return 0;
15 }
16
```

8. Print sum of odd numbers from in given range.

```
#include <stdio.h>
```

```
int main() {  
    int start, end, sum = 0;  
  
    printf("Enter the range (start end): ");  
  
    scanf("%d %d", &start, &end);  
  
    for (int i = start; i <= end; i++) {  
        if (i % 2 != 0) {  
            sum += i;  
        }  
    }  
    cout << "Sum of odd numbers: " << sum << endl;  
}
```

```
    }  
  
}  
  
printf("Sum of odd numbers in the range %d to %d: %d\n", start, end, sum);  
  
return 0;  
  
}
```

A screenshot of the Visual Studio Code (VS Code) interface. The top navigation bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar contains icons for file operations like Open, Save, Find, and others. The main editor area shows a C program named 'problem1.c' with the following code:

```
1 #include <stdio.h>
2
3 int main() {
4     int start, end, sum = 0;
5     printf("Enter the range (start end): ");
6     scanf("%d %d", &start, &end);
7
8     for (int i = start; i <= end; i++) {
9         if (i % 2 != 0) {
10             sum += i;
11         }
12     }
13
14     printf("Sum of odd numbers in the range %d to %d: %d\n", start, end, sum);
15
16     return 0;
}
```

The bottom navigation bar has tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS. The TERMINAL tab is active, showing a command-line session:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if (?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter the range (start end): 1 2
Sum of odd numbers in the range 1 to 2: 1
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

9.Base salary of a person is 50000.and he also get 40% bonus of house rent and 15% bonus of other.print gross salary.

```
#include <stdio.h>
```

```
int main() {  
    float baseSalary = 50000;  
    float houseRentBonus = 0.4 * baseSalary;  
    float otherBonus = 0.15 * baseSalary;
```

```
float grossSalary = baseSalary + houseRentBonus + otherBonus;
```

```
printf("Gross Salary: %.2f\n", grossSalary);
```

```
return 0;
```

}

A screenshot of the Visual Studio Code interface. The top bar shows the menu: File, Edit, Selection, View, Go, Run, Terminal, Help. The title bar says 'A java practise'. The left sidebar has icons for file types like Java (.java), C/C++ (.c), JavaScript (.js), and HTML (.html). A search bar at the top right contains the text 'A java practise'. The main editor area displays a C program named 'problem1.c' with the following code:

```
1 #include <stdio.h>
2
3 int main() {
4     float baseSalary = 50000;
5     float houseRentBonus = 0.4 * baseSalary;
6     float otherBonus = 0.15 * baseSalary;
7     float grossSalary = baseSalary + houseRentBonus + otherBonus;
8
9     printf("Gross Salary: %.2f\n", grossSalary);
10
11    return 0;
12 }
13
```

The bottom navigation bar includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS. The TERMINAL tab is active, showing a terminal session with the command 'cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston"' followed by the output 'Gross Salary: 77500.00'. The status bar at the bottom shows file statistics: Ln 13, Col 1 (295 selected), Spaces: 4, UTF-8, CRLF, and various live preview icons.

10. Write a program in C to display the cube of the number upto given an integer.

```
#include <stdio.h>
```

```
int main() {
```

```
int n;
```

```
printf("Enter the value of n: ");
```

```
scanf("%d", &n);
```

```
printf("Cubes of numbers up to %d are:\n", n);
```

```
for (int i = 1; i <= n; i++) {  
    printf("%d^3 = %d\n", i, i * i * i);  
}  
  
return 0;  
}
```

The screenshot shows a code editor with a terminal below it. The terminal window has a title bar 'A java practise' and displays a command-line session. The code in the editor is a C program named 'problem1.c' that prompts the user for a value of n and then prints the cubes of all numbers from 1 to n.

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the value of n: ");
    scanf("%d", &n);
    printf("Cubes of numbers up to %d are:\n", n);
    for (int i = 1; i <= n; i++) {
        printf("%d^3 = %d\n", i, i * i * i);
    }
    return 0;
}
```

TERMINAL

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter the value of n: 60
Cubes of numbers up to 60 are:
1^3 = 1
2^3 = 8
3^3 = 27
4^3 = 64
5^3 = 125
6^3 = 216
7^3 = 343
8^3 = 512
9^3 = 729
```

11. Write a program in C to display the n terms of odd natural number and their sum.

```
#include <stdio.h>
```

```
int main() {  
    int n, sum = 0;  
  
    printf("Enter the value of n: ");  
  
    scanf("%d", &n);  
  
    printf("Odd natural numbers up to %d are:\n", n);  
}
```

```
for (int i = 1; i <= n; i += 2) {  
    printf("%d ", i);  
    sum += i;  
}  
  
printf("\nSum of odd natural numbers up to %d: %d\n", n, sum);  
  
return 0;  
}
```

A screenshot of the Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar containing "A java practise". The left sidebar features icons for file operations like Open, Save, Find, and Settings. The main workspace shows a Java file named "problem1.c" with the following code:

```
5 printf("Enter the value of n: ");
6 scanf("%d", &n);
7
8 printf("Odd natural numbers up to %d are:\n", n);
9 for (int i = 1; i <= n; i += 2) {
10     printf("%d ", i);
11     sum += i;
12 }
13
14 printf("\nSum of odd natural numbers up to %d: %d\n", n, sum);
15
16 return 0;
17 }
```

The "TERMINAL" tab is selected at the bottom, displaying the output of running the code in a terminal window:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter the value of n: 33
Odd natural numbers up to 33 are:
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31 33
Sum of odd natural numbers up to 33: 289
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

12. Write a program in C to display the pattern like right angle triangle with a number. The pattern like : * * * ***

```
#include <stdio.h>

int main() {
    int n;
    printf("Enter the number of rows: ");
}
```

```

scanf("%d", &n);

for (int i = 1; i <= n; i++) {
    for (int j = 1; j <= i; j++) {
        printf("* ");
    }
    printf("\n");
}

return 0;
}

```

The screenshot shows a code editor interface with a dark theme. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. The title bar says "A java practise". The left sidebar has icons for various files: Calculation.java, cal1.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (which is the active file), Car.java, and Student.java. The main code area contains the following C code:

```

1 #include <stdio.h>
2
3 int main() {
4     int n;
5     printf("Enter the number of rows: ");
6     scanf("%d", &n);
7
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= i; j++) {
10            printf("* ");
11        }
12        printf("\n");
13    }
14}

```

Below the code, the terminal window displays the command-line interface. It shows the user navigating to a directory named "oop\Capston", running a compiler (gcc), and then executing the generated binary. The user enters "3" when prompted for the number of rows. The output is a right-angled triangle pattern of three rows:

```

PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter the number of rows: 3
*
* *
* * *

```

The bottom status bar shows system information like battery level (0.74%), network (USD/GBP +0.74%), and system time (9:17 PM, 11/15/2023).

13. Write a program in C to make such a pattern like right angle triangle with number increased by 1. The pattern like : 1 2 3 4 5 6 7 8 9 10

```
#include <stdio.h>
```

```

int main() {

    int n, num = 1;

    printf("Enter the number of rows: ");

    scanf("%d", &n);

    for (int i = 1; i <= n; i++) {

        for (int j = 1; j <= i; j++) {

            printf("%d ", num++);

        }

        printf("\n");

    }

}

return 0;
}

```

The screenshot shows a code editor interface with multiple tabs open. The active tab is 'problem1.c'. The code in the editor is:

```

1 #include <stdio.h>
2
3 int main() {
4     int n, num = 1;
5     printf("Enter the number of rows: ");
6     scanf("%d", &n);
7
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= i; j++) {
10             printf("%d ", num++);
11         }
12         printf("\n");
13     }
14
15     return 0;
16 }
17

```

Below the code editor is a terminal window showing the command-line interface. The user runs 'gcc' to compile the code and then executes the compiled binary. The output is a triangular pattern of numbers:

```

PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { ./tempCodeRunnerFile }
Enter the number of rows: 5
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15

```

14. Write a program in C to display the pattern like a diamond. * *** ***** ***** * **** *

```
#include <stdio.h>

int main() {
    int n;
    printf("Enter the number of rows (odd): ");
    scanf("%d", &n);

    for (int i = 1; i <= n; i += 2) {
        for (int space = 0; space < (n - i) / 2; space++) {
            printf(" ");
        }
        for (int j = 1; j <= i; j++) {
            printf("*");
        }
        printf("\n");
    }

    for (int i = n - 2; i >= 1; i -= 2) {
        for (int space = 0; space < (n - i) / 2; space++) {
            printf(" ");
        }
        for (int j = 1; j <= i; j++) {
            printf("*");
        }
        printf("\n");
    }
}
```

```
return 0;
```

}

15. Write a C Program to display the pattern like pyramid using the alphabet. A A B A A B C B A A B C D C B A

```
#include <stdio.h>
```

```
int main() {
```

```
int n;
```

```
printf("Enter the number of rows: ");
```

```
scanf("%d", &n);
```

```
for (int i = 1; i <= n; i++) {
```

```
char ch = 'A';
```

```
for (int j = 1; j <= i; j++) {
```

```
printf("%c ", ch++);
```

}

```
    printf("\n");
```

```
}
```

```
return 0;
```

```
}
```

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);
    for (int i = 1; i <= n; i++) {
        char ch = 'A';
        for (int j = 1; j <= i; j++) {
            printf("%c ", ch++);
        }
        printf("\n");
    }
}
```

PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if (?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }

Enter the number of rows: 6

A
A B
A B C
A B C D
A B C D E
A B C D E F

PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>

16. Write a C program to find whether a given year is a leap year or not.

```
#include <stdio.h>
```

```
int main() {
```

```
    int year;
```

```
    printf("Enter a year: ");
```

```
    scanf("%d", &year);
```

```
    if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) {
```

```
        printf("%d is a leap year.\n", year);
```

```
    } else {  
        printf("%d is not a leap year.\n", year);  
    }  
  
    return 0;  
}
```

A screenshot of the Visual Studio Code interface. The top navigation bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar contains icons for file operations like Open, Save, Find, and Settings. The main editor area shows a C program named 'problem1.c' with code for determining if a given year is a leap year. Below the editor is a terminal window showing the execution of the program. The bottom status bar displays system information such as battery level (27% Haze), network (WIFI), and system details (Ln 6, Col 24, etc.).

```
1 #include <stdio.h>
2
3 int main() {
4     int year;
5     printf("Enter a year: ");
6     scanf("%d", &year);
7
8     if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) {
9         printf("%d is a leap year.\n", year);
10    } else {
11        printf("%d is not a leap year.\n", year);
12    }
13
14    return 0;
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if (?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter a year: 2023
2023 is not a leap year.
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

17. Write a C program to find the largest of three numbers.

```
#include <stdio.h>
```

```
int main() {  
    int num1, num2, num3;  
  
    printf("Enter three numbers: ");  
  
    scanf("%d %d %d", &num1, &num2, &num3);  
  
    if (num1 >= num2 && num1 >= num3) {
```

```

printf("%d is the largest.\n", num1);

} else if (num2 >= num1 && num2 >= num3) {

    printf("%d is the largest.\n", num2);

} else {

    printf("%d is the largest.\n", num3);

}

return 0;
}

```

```

File Edit Selection View Go Run Terminal Help ← → A java practise
J Calculation.java J cal1.java J Mycalculation.java J Vehicle.java J problem2.java JS test.js O test.html C problem1.c X J Car.java J Student.java D ...
oop > Capston > C problem1.c ⚡ main()
1 #include <stdio.h>
2
3 int main() {
4     int num1, num2, num3;
5     printf("Enter three numbers: ");
6     scanf("%d %d %d", &num1, &num2, &num3);
7
8     if (num1 >= num2 && num1 >= num3) {
9         printf("%d is the largest.\n", num1);
10    } else if (num2 >= num1 && num2 >= num3) {
11        printf("%d is the largest.\n", num2);
12    } else {
13        printf("%d is the largest.\n", num3);
14    }
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter three numbers: 5
2 3
5 is the largest.
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>

```

The screenshot shows a Visual Studio Code interface with a dark theme. The top bar includes tabs for various files like Calculation.java, cal1.java, etc., and a terminal tab labeled 'A java practise'. The main editor area contains a C program that prompts for three integers and prints the largest one. Below the editor is a terminal window showing the execution of the program and its output. The bottom status bar displays file paths, line and column numbers, and system information.

18.. Write a C program to read temperature in centigrade and display a suitable message according to temperature state below. Temp < 0 then Freezing weather Temp 0-10 then Very Cold weather Temp 10-20 then Cold weather Temp 20-30 then Normal in Temp Temp 30-40 then Its Hot Temp >=40 then Its Very Hot
Test Data : 42 Expected Output : Its very hot

```
#include <stdio.h>
```

```
int main() {  
    float temperature;  
  
    printf("Enter the temperature in centigrade: ");  
    scanf("%f", &temperature);  
  
  
    if (temperature < 0) {  
        printf("Freezing weather\n");  
    } else if (temperature >= 0 && temperature <= 10) {  
        printf("Very Cold weather\n");  
    } else if (temperature > 10 && temperature <= 20) {  
        printf("Cold weather\n");  
    } else if (temperature > 20 && temperature <= 30) {  
        printf("Normal in Temp\n");  
    } else if (temperature > 30 && temperature <= 40) {  
        printf("It's Hot\n");  
    } else {  
        printf("It's Very Hot\n");  
    }  
  
    return 0;  
}
```

```
5 printf("Enter the temperature in centigrade: ");
6 scanf("%f", &temperature);
7
8 if (temperature < 0) {
9     printf("Freezing weather\n");
10 } else if (temperature >= 0 && temperature <= 10) {
11     printf("Very Cold weather\n");
12 } else if (temperature > 10 && temperature <= 20) {
13     printf("Cold weather\n");
14 } else if (temperature > 20 && temperature <= 30) {
15     printf("Normal in Temp\n");
16 } else if (temperature > 30 && temperature <= 40) {
17     printf("It's Hot\n");
18 }
19
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter the temperature in centigrade: 29
Normal in Temp
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

19. Write a C program to check whether a triangle can be formed by the given value for the angles.

```
#include <stdio.h>

int main() {

    int angle1, angle2, angle3;

    printf("Enter three angles of the triangle: ");

    scanf("%d %d %d", &angle1, &angle2, &angle3);

    if (angle1 + angle2 + angle3 == 180) {

        printf("Triangle can be formed with these angles.\n");

    } else {

        printf("Triangle cannot be formed with these angles.\n");

    }

}
```

```
return 0;
```

```
}
```

The screenshot shows a code editor interface with a dark theme. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. Below the menu is a tab bar with several files: Calculation.java, cal1.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (the active file), Car.java, and Student.java. The main workspace contains the following C code:

```
1 #include <stdio.h>
2
3 int main() {
4     int angle1, angle2, angle3;
5
6     printf("Enter three angles of the triangle: ");
7     scanf("%d %d %d", &angle1, &angle2, &angle3);
8
9     if (angle1 + angle2 + angle3 == 180) {
10         printf("Triangle can be formed with these angles.\n");
11     } else {
12         printf("Triangle cannot be formed with these angles.\n");
13     }
14 }
```

Below the code, a terminal window displays the following command-line interaction:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter three angles of the triangle: 5 6 9
Triangle cannot be formed with these angles.
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The bottom of the screen shows a taskbar with various icons and system status information.

20. Write a C program to check whether a character is an alphabet, digit or special character.

```
#include <stdio.h>
```

```
int main() {
    char ch;

    printf("Enter a character: ");

    scanf(" %c", &ch);

    if ((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z')) {
        printf("%c is an alphabet.\n", ch);
    } else if (ch >= '0' && ch <= '9') {
        printf("%c is a digit.\n", ch);
    }
}
```

```

} else {

    printf("%c is a special character.\n", ch);

}

return 0;
}

```

```

File Edit Selection View Go Run Terminal Help ← → A java practise
J Calculation.java J call.java J Mycalculation.java J Vehicle.java J problem2.java JS test.js O test.html C problem1.c X J Car.java J Student.java D v ⚙ ...
oop > Capston > C problem1.c > main()
1 #include <stdio.h>
2
3 int main() {
4     char ch;
5
6     printf("Enter a character: ");
7     scanf(" %c", &ch);
8
9     if ((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z')) {
10         printf("%c is an alphabet.\n", ch);
11     } else if (ch >= '0' && ch <= '9') {
12         printf("%c is a digit.\n", ch);
13     } else {
14         printf("%c is a special character.\n", ch);
}
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter a character: a
a is an alphabet.
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>

```

The screenshot shows a Visual Studio Code interface with a dark theme. The left sidebar has icons for file operations like copy, paste, and search. The top bar includes tabs for various files and a search bar. The main area displays a C program with syntax highlighting. The terminal at the bottom shows the program's execution and output. The taskbar at the bottom of the screen shows other open applications like a browser and file explorer.

21. Write a C program to check whether an alphabet is a vowel or consonant.

```
#include <stdio.h>
```

```

int main() {

char ch;

printf("Enter an alphabet: ");

scanf(" %c", &ch);
}

```

```

if ((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z')) {

    if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u' ||
        ch == 'A' || ch == 'E' || ch == 'I' || ch == 'O' || ch == 'U') {

        printf("%c is a vowel.\n", ch);

    } else {

        printf("%c is a consonant.\n", ch);

    }

} else {

    printf("Invalid input.\n");

}

}

return 0;
}

```

```

File Edit Selection View Go Run Terminal Help ← → 🔍 A Java practise
File Edit Selection View Go Run Terminal Help ← → 🔍 A Java practise
Calculation.java J call.java J Mycalculation.java J Vehicle.java J problem2.java JS test.js ⌂ test.html C problem1.c X J Car.java J Student.java
oop > Capston > C problem1.c > main()
3 int main() {
4     char ch;
5
6     printf("Enter an alphabet: ");
7     scanf(" %c", &ch);
8
9     if ((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z')) {
10         if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u' ||
11             ch == 'A' || ch == 'E' || ch == 'I' || ch == 'O' || ch == 'U') {
12             printf("%c is a vowel.\n", ch);
13         } else {
14             printf("%c is a consonant.\n", ch);
15         }
16     }
17 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if (?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter an alphabet: A
A is a vowel.
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>

```

22. Write a program in C to read any day number in integer and display day name in the word.

```
#include <stdio.h>
```

```
int main() {  
  
    int dayNumber;  
  
    printf("Enter the day number (1-7): ");  
    scanf("%d", &dayNumber);  
  
    switch (dayNumber) {  
  
        case 1:  
            printf("Sunday\n");  
            break;  
  
        case 2:  
            printf("Monday\n");  
            break;  
  
        case 3:  
            printf("Tuesday\n");  
            break;  
  
        case 4:  
            printf("Wednesday\n");  
            break;  
  
        case 5:  
            printf("Thursday\n");  
            break;  
  
        case 6:  
            printf("Friday\n");  
            break;  
  
        case 7:  
    }  
}
```

```

printf("Saturday\n");

break;

default:

printf("Invalid day number.\n");

}

return 0;
}

```

```

File Edit Selection View Go Run Terminal Help ← → A java practise
File Calculation.java J cal1.java J Mycalculation.java J Vehicle.java J problem2.java JS test.js D test.html C problem1.c X J Car.java J Student.java D ... E X
oop > Capston > C problem1.c > main()
3 int main() {
4     int dayNumber;
5
6     printf("Enter the day number (1-7): ");
7     scanf("%d", &dayNumber);
8
9     switch (dayNumber) {
10         case 1:
11             printf("Sunday\n");
12             break;
13         case 2:
14             printf("Monday\n");
15             break;
16         case 3:
17             printf("Tuesday\n");
18             break;
19         case 4:
20             printf("Wednesday\n");
21             break;
22         case 5:
23             printf("Thursday\n");
24             break;
25         case 6:
26             printf("Friday\n");
27             break;
28         case 7:
29             printf("Saturday\n");
30             break;
31     }
32 }
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
279
280
281
282
283
284
285
286
287
288
289
289
290
291
292
293
294
295
296
297
298
299
299
300
301
302
303
304
305
306
307
308
309
309
310
311
312
313
314
315
316
317
318
319
319
320
321
322
323
324
325
326
327
328
329
329
330
331
332
333
334
335
336
337
338
339
339
340
341
342
343
344
345
346
347
348
349
349
350
351
352
353
354
355
356
357
358
359
359
360
361
362
363
364
365
366
367
368
369
369
370
371
372
373
374
375
376
377
378
379
379
380
381
382
383
384
385
386
387
388
389
389
390
391
392
393
394
395
396
397
398
399
399
400
401
402
403
404
405
406
407
408
409
409
410
411
412
413
414
415
416
417
418
419
419
420
421
422
423
424
425
426
427
428
429
429
430
431
432
433
434
435
436
437
438
439
439
440
441
442
443
444
445
446
447
448
449
449
450
451
452
453
454
455
456
457
458
459
459
460
461
462
463
464
465
466
467
468
469
469
470
471
472
473
474
475
476
477
478
479
479
480
481
482
483
484
485
486
487
488
489
489
490
491
492
493
494
495
496
497
498
499
499
500
501
502
503
504
505
506
507
508
509
509
510
511
512
513
514
515
516
517
518
519
519
520
521
522
523
524
525
526
527
528
529
529
530
531
532
533
534
535
536
537
538
539
539
540
541
542
543
544
545
546
547
548
549
549
550
551
552
553
554
555
556
557
558
559
559
560
561
562
563
564
565
566
567
568
569
569
570
571
572
573
574
575
576
577
578
579
579
580
581
582
583
584
585
586
587
588
589
589
590
591
592
593
594
595
596
597
598
599
599
600
601
602
603
604
605
606
607
608
609
609
610
611
612
613
614
615
616
617
618
619
619
620
621
622
623
624
625
626
627
628
629
629
630
631
632
633
634
635
636
637
638
639
639
640
641
642
643
644
645
646
647
648
649
649
650
651
652
653
654
655
656
657
658
659
659
660
661
662
663
664
665
666
667
668
669
669
670
671
672
673
674
675
676
677
678
679
679
680
681
682
683
684
685
686
687
688
689
689
690
691
692
693
694
695
696
697
697
698
699
699
700

```

23. Write a program in C to read any Month Number in integer and display Month name in the word.

```
#include <stdio.h>
```

```

int main() {

int monthNumber;

printf("Enter the month number (1-12): ");

```

```
scanf("%d", &monthNumber);
```

```
switch (monthNumber) {
```

```
    case 1:
```

```
        printf("January\n");
```

```
        break;
```

```
    case 2:
```

```
        printf("February\n");
```

```
        break;
```

```
    case 3:
```

```
        printf("March\n");
```

```
        break;
```

```
    case 4:
```

```
        printf("April\n");
```

```
        break;
```

```
    case 5:
```

```
        printf("May\n");
```

```
        break;
```

```
    case 6:
```

```
        printf("June\n");
```

```
        break;
```

```
    case 7:
```

```
        printf("July\n");
```

```
        break;
```

```
    case 8:
```

```
        printf("August\n");
```

```
        break;
```

```
case 9:
```

```
    printf("September\n");
```

```
    break;
```

```
case 10:
```

```
    printf("October\n");
```

```
    break;
```

```
case 11:
```

```
    printf("November\n");
```

```
    break;
```

```
case 12:
```

```
    printf("December\n");
```

```
    break;
```

```
default:
```

```
    printf("Invalid month number.\n");
```

```
}
```

```
return 0;
```

```
}
```

The screenshot shows a code editor interface with a dark theme. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar has icons for various file types. The main editor area contains a C program:

```
oop > Capston > C problem1.c > main()
5
6     printf("Enter the month number (1-12): ");
7     scanf("%d", &monthNumber);
8
9     switch (monthNumber) {
10         case 1:
11             printf("January\n");
12             break;
13         case 2:
14             printf("February\n");
15             break;
16         case 3:
17             printf("March\n");
18             break;
```

The terminal below shows the command to run the program and its output:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter the month number (1-12): 7
July
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The bottom status bar shows the current file path, line count (Ln 16, Col 16), spaces (Spaces: 4), encoding (UTF-8), and other system information.

24. Write a program in C to read any Month Number in integer and display the number of days for this month.

```
#include <stdio.h>
```

```
int main() {
    int monthNumber;

    printf("Enter the month number (1-12): ");
    scanf("%d", &monthNumber);
```

```
switch (monthNumber) {
```

```
    case 1:
```

```
    case 3:
```

```
    case 5:
```

```
    case 7:
```

```
    case 8:
```

```
case 10:  
  
case 12:  
  
    printf("31 days\n");  
  
    break;
```

```
case 4:
```

```
case 6:
```

```
case 9:
```

```
case 11:
```

```
    printf("30 days\n");  
  
    break;
```

```
case 2:
```

```
    printf("28 or 29 days (leap year)\n");  
  
    break;
```

```
default:
```

```
    printf("Invalid month number.\n");  
  
}
```

```
return 0;
```

```
}
```

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar has icons for file types like Java (.java), JavaScript (.js), and HTML (.html). The main editor window displays a C program named 'problem1.c' with code for calculating days in a month based on the input month number (1-12). The terminal below shows the command to compile and run the program, followed by the output '31 days'. The status bar at the bottom provides system information like temperature (27°C Haze), battery level (0%), and system details (Ln 16, Col 17, Spaces: 4, UTF-8, CRLF).

```
oop > Capston > C problem1.c > main()
22     printf("30 days\n");
23     break;
24     case 2:
25         printf("28 or 29 days (leap year)\n");
26         break;
27     default:
28         printf("Invalid month number.\n");
29     }
30 }
31
32     return 0;
33 }
34
```

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter the month number (1-12): 12
31 days
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

25. Write a program that converts Centigrade to Kelvin

```
#include <stdio.h>
```

```
int main() {
    float celsius, kelvin;

    printf("Enter temperature in Centigrade: ");
    scanf("%f", &celsius);
```

```
    kelvin = celsius + 273.15;
```

```
    printf("Temperature in Kelvin: %.2f K\n", kelvin);
```

```
    return 0;
```

```
}
```

A screenshot of Microsoft Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar shows several open files: Calculation.java, call.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (which is the active tab), Car.java, and Student.java. The main code editor window displays the following C code:

```
1 #include <stdio.h>
2
3 int main() {
4     float celsius, kelvin;
5
6     printf("Enter temperature in Centigrade: ");
7     scanf("%f", &celsius);
8
9     kelvin = celsius + 273.15;
10
11    printf("Temperature in Kelvin: %.2f K\n", kelvin);
12
13    return 0;
14 }
```

The terminal below the editor shows the command-line output of running the program:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter temperature in Centigrade: 63
Temperature in Kelvin: 336.15 K
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The bottom status bar shows various system icons and information like the date and time.

1. Write a C program to print your name, date of birth, and mobile number.

```
#include <stdio.h>
```

```
int main() {

    printf("Name: Your Name\n");

    printf("Date of Birth: Your DOB\n");

    printf("Mobile Number: Your Mobile Number\n");



    return 0;
}
```

```
#include <stdio.h>
int main() {
    printf("Name: Your Name\n");
    printf("Date of Birth: Your DOB\n");
    printf("Mobile Number: Your Mobile Number\n");
    return 0;
}
```

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Name: Your Name
Date of Birth: Your DOB
Mobile Number: Your Mobile Number
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

2. Write a C program to print the following characters in a reverse way. Test Characters: 'X', 'M', 'L' Expected Output: The reverse of XML is LMX

```
#include <stdio.h>
```

```
int main() {
    char chars[] = {'X', 'M', 'L'};
    printf("Test Characters: '%c', '%c', '%c'\n", chars[0], chars[1], chars[2]);
    printf("The reverse of XML is %c%c%c\n", chars[2], chars[1], chars[0]);
    return 0;
}
```

The screenshot shows a Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar has icons for file types like Java, C, and JavaScript. The main editor tab is 'C problem1.c'. The code in the editor is:

```
1 #include <stdio.h>
2
3 int main() {
4     char chars[] = {'X', 'M', 'L'};
5     printf("Test Characters: %c, %c, %c\n", chars[0], chars[1], chars[2]);
6     printf("The reverse of XML is %c%c%c\n", chars[2], chars[1], chars[0]);
7
8     return 0;
9 }
10
```

The terminal below shows the command line output:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Test Characters: 'X', 'M', 'L'
The reverse of XML is LMX
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The bottom status bar shows system information like battery level (27°C Haze), network, and date/time (11/15/2023 9:35 PM).

3. Write a C program to convert specified days into years, weeks and days.

```
#include <stdio.h>
```

```
int main() {
    int days;

    printf("Enter the number of days: ");
    scanf("%d", &days);
```

```
    int years = days / 365;
    int weeks = (days % 365) / 7;
    int remainingDays = (days % 365) % 7;
```

```
    printf("%d days is equal to %d years, %d weeks, and %d days.\n", days, years, weeks, remainingDays);
```

```
return 0;
```

```
}
```

```
File Edit Selection View Go Run Terminal Help A java practise
J Calculation.java J cal1.java J Mycalculation.java J Vehicle.java J problem2.java JS test.js O test.html C problem1.c X J Car.java J Student.java
oop > Capston > C problem1.c ...
5
6     ... printf("Enter the number of days: ");
7     ... scanf("%d", &days);
8
9     ... int years = days / 365;
10    ... int weeks = (days % 365) / 7;
11    ... int remainingDays = (days % 365) % 7;
12
13    ... printf("%d days is equal to %d years, %d weeks, and %d days.\n", days, years, weeks, remainingDays);
14
15    ... return 0;
16 }
17
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter the number of days: 5
5 days is equal to 0 years, 0 weeks, and 5 days.
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
27°C Haze
```

Search

Ln 17, Col 1 (358 selected) Spaces: 4 UTF-8 CRLF ⚡ Go Live ⚡ Port: 5500 Win32 ⚡ Go Live ⚡ 9:36 PM 11/15/2023

4. Write a C program to calculate the distance between the two points.

```
#include <stdio.h>
```

```
#include <math.h>
```

```
int main() {
```

```
    float x1, y1, x2, y2, distance;
```

```
    printf("Enter the coordinates of point 1 (x1 y1): ");
```

```
    scanf("%f %f", &x1, &y1);
```

```
    printf("Enter the coordinates of point 2 (x2 y2): ");
```

```
    scanf("%f %f", &x2, &y2);
```

```
distance = sqrt(pow(x2 - x1, 2) + pow(y2 - y1, 2));
```

```
printf("Distance between the two points: %.2f\n", distance);
```

```
return 0;
```

```
}
```

The screenshot shows a Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar. The title bar says "A java practise". The left sidebar has icons for file types like Java (.java), JavaScript (.js), and C/C++ (.c). The main editor area contains the following C code:

```
3
4 int main() {
5     float x1, y1, x2, y2, distance;
6
7     printf("Enter the coordinates of point 1 (x1 y1): ");
8     scanf("%f %f", &x1, &y1);
9
10    printf("Enter the coordinates of point 2 (x2 y2): ");
11    scanf("%f %f", &x2, &y2);
12
13    distance = sqrt(pow(x2 - x1, 2) + pow(y2 - y1, 2));
14
15    printf("Distance between the two points: %.2f\n", distance);
16}
```

The terminal below the editor shows the command-line output:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { ./tempCodeRunnerFile }
Enter the coordinates of point 1 (x1 y1): 1 2
Enter the coordinates of point 2 (x2 y2): 9 6
Distance between the two points: 8.94
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The bottom status bar shows the terminal settings: Line 8, Column 30, Spaces: 4, UTF-8, CRLF, and the date/time: 9:37 PM 11/15/2023.

5. Write a C program to calculate the value of S where $S = 1 + 1/2 + 1/3 + \dots + 1/50$.

```
#include <stdio.h>
```

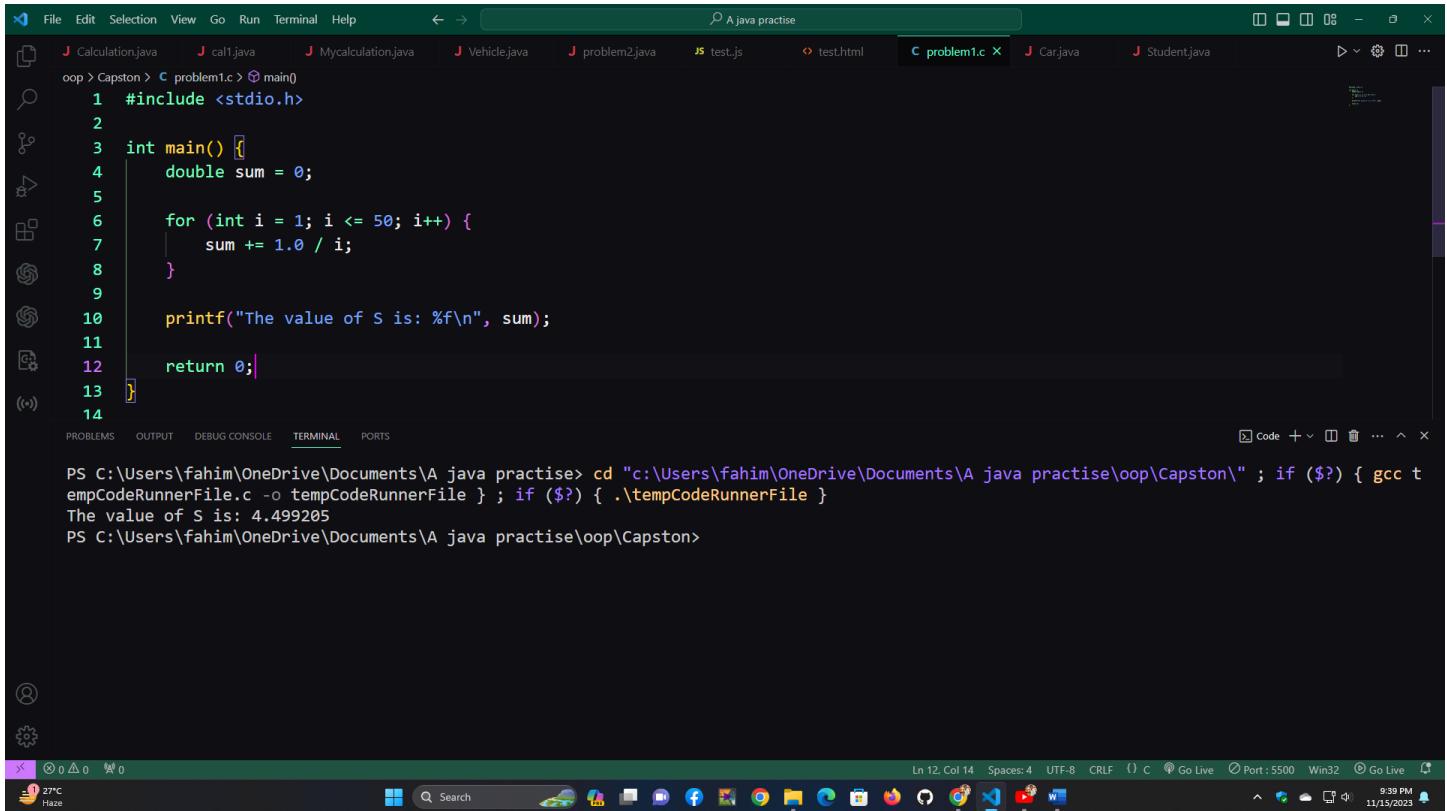
```
int main() {
    double sum = 0;

    for (int i = 1; i <= 50; i++) {
        sum += 1.0 / i;
    }
}
```

```
printf("The value of S is: %f\n", sum);
```

```
return 0;
```

```
}
```



```
#include <stdio.h>
int main() {
    double sum = 0;
    for (int i = 1; i <= 50; i++) {
        sum += 1.0 / i;
    }
    printf("The value of S is: %f\n", sum);
    return 0;
}
```

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
The value of S is: 4.499205
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

6. Write a C program to accept two integers and check whether they are equal or not.

```
#include <stdio.h>
```

```
int main() {
```

```
    int num1, num2;
```

```
    printf("Enter two integers: ");
```

```
    scanf("%d %d", &num1, &num2);
```

```
    if (num1 == num2) {
```

```
    printf("The two numbers are equal.\n");  
}  
else {  
    printf("The two numbers are not equal.\n");  
}  
  
return 0;  
}
```

The screenshot shows a Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar contains icons for file operations like Open, Save, Find, and others. The main editor area displays a C program named 'problem1.c' with the following code:

```
1 #include <stdio.h>
2
3 int main() {
4     int num1, num2;
5
6     printf("Enter two integers: ");
7     scanf("%d %d", &num1, &num2);
8
9     if (num1 == num2) {
10         printf("The two numbers are equal.\n");
11     } else {
12         printf("The two numbers are not equal.\n");
13     }
14 }
```

Below the editor, there are tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS. The TERMINAL tab is active, showing a terminal session with the following output:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter two integers: 1 2
The two numbers are not equal.
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The bottom status bar shows system information including the date and time (9:40 PM 11/15/2023), and various icons for system monitoring and connectivity.

7. Write a C program to check whether a given number is positive or negative.

```
#include <stdio.h>
```

```
int main() {  
    int num;  
  
    printf("Enter a number: ");  
  
    scanf("%d", &num);
```

```

if (num > 0) {

    printf("The number is positive.\n");

} else if (num < 0) {

    printf("The number is negative.\n");

} else {

    printf("The number is zero.\n");
}

```

```
return 0;
```

```
}
```

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The top bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar. The title bar says "A java practise". The left sidebar has icons for file, folder, search, and other development tools. The main editor area contains a C program:

```

#include <stdio.h>
int main() {
    int num;
    printf("Enter a number: ");
    scanf("%d", &num);
    if (num > 0) {
        printf("The number is positive.\n");
    } else if (num < 0) {
        printf("The number is negative.\n");
    } else {
        printf("The number is zero.\n");
    }
}

```

The status bar at the bottom shows the terminal command: PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if (\$?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (\$?) { ./tempCodeRunnerFile }

The bottom right corner shows system status: 27°C Haze, 942 PM, 11/15/2023.

8. Write a C program that calculates the volume of a sphere.

```
#include <stdio.h>

#include <math.h>
```

```
int main() {  
    double radius, volume;  
  
    printf("Enter the radius of the sphere: ");  
    scanf("%lf", &radius);  
  
    volume = (4.0 / 3.0) * M_PI * pow(radius, 3);  
  
    printf("Volume of the sphere: %.2f\n", volume);  
  
    return 0;  
}
```

A screenshot of the Visual Studio Code (VS Code) interface. The top bar shows the file menu (File, Edit, Selection, View, Go, Run, Terminal, Help), a back/forward navigation bar, a search bar containing "A java practise", and a system tray with icons for battery, signal, and time (9:46 PM, 11/15/2023).

The left sidebar contains icons for various file types: Calculation.java, cal1.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (the active file), Car.java, and Student.java.

The main editor area displays a C program named "problem1.c". The code prompts the user for the radius of a sphere, calculates its volume using the formula $(4.0 / 3.0) * 3.1416 * \text{radius}^3$, and prints the result. The code uses standard input/output headers and functions from the math library.

Below the editor, tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS are visible. The TERMINAL tab is selected, showing a terminal session where the program is run and its output is displayed. The output shows the command to change directory to "Capston", compilation of "tempCodeRunnerFile.c" to "tempCodeRunnerFile", and the execution of "tempCodeRunnerFile". The user then enters "6" as the radius, and the program outputs "Volume of the sphere: 904.78".

The bottom status bar shows system information: battery level (27%), signal strength (Haze), and file details (Ln 12, Col 52, Spaces: 4, UTF-8, CRLF). It also includes icons for Go Live, Port: 5500, Win32, and Go Live, along with a taskbar with various application icons.

9. Write a C program to find the third angle of a triangle if two angles are given.

```
#include <stdio.h>
```

```

int main() {

    int angle1, angle2, angle3;

    printf("Enter the first angle: ");

    scanf("%d", &angle1);

    printf("Enter the second angle: ");

    scanf("%d", &angle2);

    angle3 = 180 - angle1 - angle2;

    printf("The third angle is: %d\n", angle3);

    return 0;
}

```

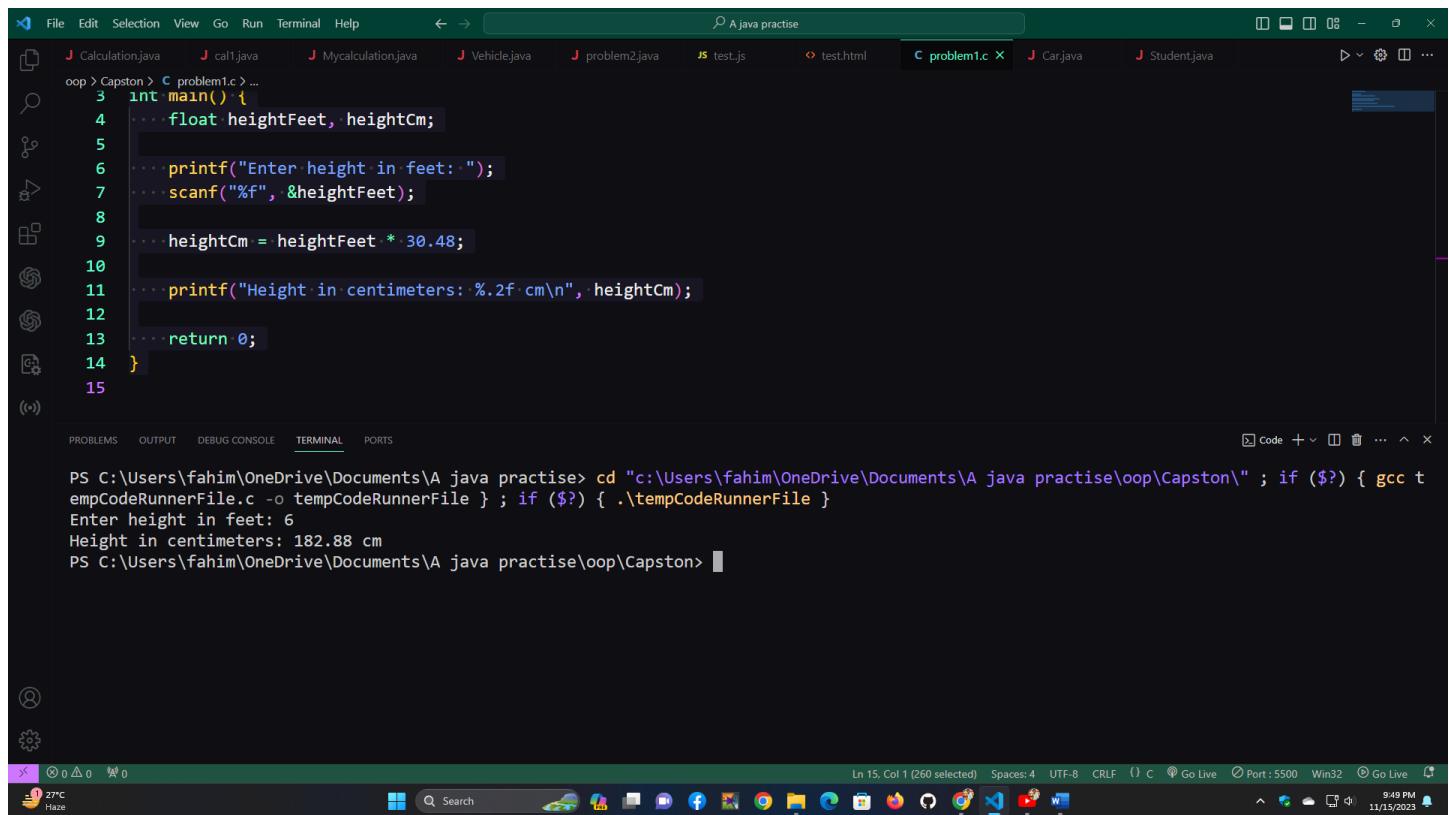
The screenshot shows a code editor interface with the following details:

- File Explorer:** Shows various files including Calculation.java, cal1.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (the current file), Car.java, and Student.java.
- Code Editor:** Displays the C code for calculating the third angle of a triangle. The code uses standard input/output functions like printf and scanf to get user input for two angles and calculate the third.
- Terminal:** A terminal window at the bottom shows the command-line execution of the program. It starts by navigating to the directory containing the C file, then compiles it with gcc, and finally runs the executable. The output shows the user entering two angles (3 and 6) and the program printing the third angle (171).
- Bottom Bar:** Includes icons for search, file operations, and system status.

10. Write a C program to convert height feet to centimetre.

```
#include <stdio.h>
```

```
int main() {  
  
    float heightFeet, heightCm;  
  
    printf("Enter height in feet: ");  
  
    scanf("%f", &heightFeet);  
  
    heightCm = heightFeet * 30.48;  
  
    printf("Height in centimeters: %.2f cm\n", heightCm);  
  
    return 0;  
}
```



```
File Edit Selection View Go Run Terminal Help ← → 🔍 A java practise  
J Calculation.java J cal1.java J Mycalculation.java J Vehicle.java J problem2.java JS test.js ⌂ test.html C problem1.c X J Carjava J Student.java ⌂ ...  
oop > Capston > C problem1.c ...  
3 int main() {  
4     float heightFeet, heightCm;  
5  
6     printf("Enter height in feet: ");  
7     scanf("%f", &heightFeet);  
8  
9     heightCm = heightFeet * 30.48;  
10  
11    printf("Height in centimeters: %.2f cm\n", heightCm);  
12  
13    return 0;  
14 }  
15
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }  
Enter height in feet: 6  
Height in centimeters: 182.88 cm  
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston> █
```

27°C Haze

Search

Ln 15, Col 1 (260 selected) Spaces: 4 UTF-8 CRLF ⓘ C ⓘ Go Live ⓘ Port: 5500 Win32 ⓘ Go Live ⓘ 9:49 PM 11/15/2023 ⓘ

11. Write a C program to perform addition, subtraction, multiplication and division of two numbers if they are even.

```
#include <stdio.h>
```

```
int main() {
    int num1, num2, sum, diff, product;
    float quotient;
```

```
printf("Enter two numbers: ");
```

```
scanf("%d %d", &num1, &num2);
```

```
if (num1 % 2 == 0 && num2 % 2 == 0) {
```

```
    sum = num1 + num2;
```

```
    diff = num1 - num2;
```

```
    product = num1 * num2;
```

```
    quotient = (float) num1 / num2;
```

```
    printf("Sum: %d\n", sum);
```

```
    printf("Difference: %d\n", diff);
```

```
    printf("Product: %d\n", product);
```

```
    printf("Quotient: %.2f\n", quotient);
```

```
} else {
```

```
    printf("Numbers are not even. Cannot perform operations.\n");
```

```
}
```

```
return 0;
```

```
}
```

```
File Edit Selection View Go Run Terminal Help ← → ⌂ A java practise ⌂ problem1.c × ⌂ Calculation.java ⌂ cal1.java ⌂ Mycalculation.java ⌂ Vehicle.java ⌂ problem2.java ⌂ test.js ⌂ test.html ⌂ Car.java ⌂ Student.java ⌂ Capston > C problem1.c ...
3 int main() {
4     int num1, num2, sum, diff, product;
5     float quotient;
6
7     printf("Enter two numbers: ");
8     scanf("%d %d", &num1, &num2);
9
10    if (num1 % 2 == 0 && num2 % 2 == 0) {
11        sum = num1 + num2;
12        diff = num1 - num2;
13        product = num1 * num2;
14        quotient = (float) num1 / num2;
15
16        printf("Sum: %d\n", sum);
}
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc t empCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter two numbers: 6 8
Sum: 14
Difference: -2
Product: 48
Quotient: 0.75
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

12. Write a C program to calculate the average of two numbers if they are odd.

```
#include <stdio.h>
```

```
int main() {
    int num1, num2;

    printf("Enter two numbers: ");
    scanf("%d %d", &num1, &num2);

    if (num1 % 2 != 0 && num2 % 2 != 0) {
        double average = (num1 + num2) / 2.0;
        printf("Average of the two odd numbers: %.2f\n", average);
    } else {
        printf("Numbers are not both odd. Cannot calculate average.\n");
    }
}
```

}

```
return 0;
```

}

A screenshot of the Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled "A java practise". The left sidebar contains icons for file operations like Open, Save, Find, and others. The main workspace shows a C file named "problem1.c" with the following code:

```
oop > Capston > C problem1.c > main()
6   printf("Enter two numbers: ");
7   scanf("%d %d", &num1, &num2);
8
9   if (num1 % 2 != 0 && num2 % 2 != 0) {
10      double average = (num1 + num2) / 2.0;
11      printf("Average of the two odd numbers: %.2f\n", average);
12    } else {
13      printf("Numbers are not both odd. Cannot calculate average.\n");
14    }
15
16    return 0;
17 }
18
```

The terminal tab at the bottom shows the command-line output of running the program:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter two numbers: 1 3
Average of the two odd numbers: 2.00
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The bottom status bar displays system information like battery level (0%), temperature (26°C), and network connection.

13. Write a C program to calculate the division of two numbers if they are negative.

```
#include <stdio.h>
```

```
int main() {  
    int num1, num2;  
  
    printf("Enter two numbers: ");  
  
    scanf("%d %d", &num1, &num2);  
  
    if (num1 < 0 && num2 < 0) {  
        double result = (double) n  
    }  
}
```

```
    printf("Division of the two negative numbers: %.2f\n", result);

} else {

    printf("Numbers are not both negative. Cannot perform division.\n");

}

return 0;
}
```

A screenshot of the Visual Studio Code (VS Code) interface. The top bar shows the menu: File, Edit, Selection, View, Go, Run, Terminal, Help. The title bar says "A java practise". The left sidebar has icons for file types like Java (.java), C/C++ (.c), JavaScript (.js), and HTML (.html). There are also icons for search, file operations, and settings. The main editor area contains a C program named "problem1.c". The code includes #include <stdio.h>, main() function, user input for two numbers, and a conditional statement to check if both numbers are negative. The terminal below shows the command to run the program, the user entering two numbers (2 5), and the output indicating that numbers are not both negative, so division cannot be performed.

```
1 #include <stdio.h>
2
3 int main() {
4     int num1, num2;
5
6     printf("Enter two numbers: ");
7     scanf("%d %d", &num1, &num2);
8
9     if (num1 < 0 && num2 < 0) {
10         double result = (double) num1 / num2;
11         printf("Division of the two negative numbers: %.2f\n", result);
12     } else {
13         printf("Numbers are not both negative. Cannot perform division.\n");
14     }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter two numbers: 2 5
Numbers are not both negative. Cannot perform division.
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

13. Input the height and base of a triangle from console and calculate the area of the triangle.

```
#include <stdio.h>
```

```
int main() {  
    float height, base, area;
```

```
printf("Enter the height of the triangle: ");
```

```

scanf("%f", &height);

printf("Enter the base of the triangle: ");

scanf("%f", &base);

area = 0.5 * base * height;

printf("Area of the triangle: %.2f\n", area);

return 0;
}

```

The screenshot shows a code editor interface with a dark theme. The main window displays a C program. The terminal window below shows the execution of the program, including user input for height and base, and the resulting output.

```

File Edit Selection View Go Run Terminal Help ← → A java practise
J Calculation.java J cal1.java J Mycalculation.java J Vehicle.java J problem2.java JS test.js D test.html C problem1.c X J Car.java J Student.java
oop > Capston > C problem1.c > main()
1 #include <stdio.h>
2
3 int main() {
4     float height, base, area;
5
6     printf("Enter the height of the triangle: ");
7     scanf("%f", &height);
8
9     printf("Enter the base of the triangle: ");
10    scanf("%f", &base);
11
12    area = 0.5 * base * height;
13
14    printf("Area of the triangle: %.2f\n", area);

```

```

PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc t empCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter the height of the triangle: 5
Enter the base of the triangle: 6
Area of the triangle: 15.00
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>

```

14. Input two numbers from console and multiply them if 1st number less than 2nd number.

```
#include <stdio.h>
```

```
int main() {
```

```

int num1, num2;

printf("Enter two numbers: ");

scanf("%d %d", &num1, &num2);

if (num1 < num2) {

    int product = num1 * num2;

    printf("Multiplication of the two numbers: %d\n", product);

} else {

    printf("The first number is not less than the second number. Cannot perform multiplication.\n");

}

return 0;
}

```

```

#include <stdio.h>
int main() {
    int num1, num2;
    printf("Enter two numbers: ");
    scanf("%d %d", &num1, &num2);
    if (num1 < num2) {
        int product = num1 * num2;
        printf("Multiplication of the two numbers: %d\n", product);
    } else {
        printf("The first number is not less than the second number. Cannot perform multiplication.\n");
    }
}

```

PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if (\$?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (\$?) { .\tempCodeRunnerFile }

Enter two numbers: 2 6

Multiplication of the two numbers: 12

PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>

15. Input a number from console and print if the number less than 20 and grater than 5.

```
#include <stdio.h>

int main() {
    int num;

    printf("Enter a number: ");
    scanf("%d", &num);

    if (num > 5 && num < 20) {
        printf("The number is between 5 and 20.\n");
    } else {
        printf("The number is not between 5 and 20.\n");
    }

    return 0;
}
```

The screenshot shows a VS Code interface with a dark theme. The left sidebar has icons for file operations like Open, Save, Find, and Refresh. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The main editor area contains a C program:

```
#include <stdio.h>
int main() {
    int num;
    printf("Enter a number: ");
    scanf("%d", &num);
    if (num > 5 && num < 20) {
        printf("The number is between 5 and 20.\n");
    } else {
        printf("The number is not between 5 and 20.\n");
    }
}
```

Below the editor, tabs for Calculation.java, call.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (which is the active tab), Car.java, and Student.java are visible. The bottom navigation bar includes PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (which is selected), and PORTS. The terminal window shows the command line output:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter a number: 23
The number is not between 5 and 20.
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The taskbar at the bottom shows system icons for weather (26°C Haze), search, file explorer, task manager, and browser, along with the date and time (11/15/2023, 10:27 PM).

16. Input character from console and print x^2+4y+c .

```
#include <stdio.h>
```

```
int main() {
```

```
    char ch;
```

```
    printf("Enter a character: ");
```

```
    scanf(" %c", &ch);
```

```
    int x = ch;
```

```
    int y = x % 4;
```

```
    int result = x * x + 4 * y + ch;
```

```
    printf("Result: %d\n", result);
```

```
return 0;
```

}

A screenshot of the Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar containing "A java practise". The left sidebar features icons for file operations like Open, Save, Find, and Settings. The main workspace shows a Java file named "problem1.c" with the following code:

```
3 int main() {
4     char ch;
5
6     printf("Enter a character: ");
7     scanf(" %c", &ch);
8
9     int x = ch;
10    int y = x % 4;
11    int result = x * x + 4 * y + ch;
12
13    printf("Result: %d\n", result);
14
15    return 0;
16}
```

The bottom navigation bar includes PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (which is selected), and PORTS. The terminal tab shows command-line output:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter a character: s
Result: 13352
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

17. Make A multiplication table for 1.

```
#include <stdio.h>
```

```
int main() {
```

```
int num = 1;
```

```
for (int i = 1; i <= 10; i++) {
```

```
printf("%d x %d = %d\n", num, i, num * i);
```

}

```
return 0;
```

}

```
#include <stdio.h>
int main() {
    int num = 1;
    for (int i = 1; i <= 10; i++) {
        printf("%d x %d = %d\n", num, i, num * i);
    }
    return 0;
}
```

PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if (\$?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }

1 x 1 = 1
1 x 2 = 2
1 x 3 = 3
1 x 4 = 4
1 x 5 = 5
1 x 6 = 6
1 x 7 = 7
1 x 8 = 8
1 x 9 = 9
1 x 10 = 10

PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>

18. Make A multiplication table for 1 To 10.

```
#include <stdio.h>
```

```
int main() {

    for (int i = 1; i <= 10; i++) {

        printf("Multiplication table for %d:\n", i);

        for (int j = 1; j <= 10; j++) {

            printf("%d x %d = %d\n", i, j, i * j);

        }

        printf("\n");

    }

    return 0;

}
```

A screenshot of the Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar shows several open files: Calculation.java, call.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (the active tab), Car.java, and Student.java. The main editor area contains the following C code:

```
#include <stdio.h>
int main() {
    for (int i = 1; i <= 10; i++) {
        printf("Multiplication table for %d:\n", i);
        for (int j = 1; j <= 10; j++) {
            printf("%d x %d = %d\n", i, j, i * j);
        }
        printf("\n");
    }
    return 0;
}
```

The terminal below the editor shows the output of the program:

```
Multiplication table for 10:
10 x 1 = 10
10 x 2 = 20
10 x 3 = 30
10 x 4 = 40
10 x 5 = 50
10 x 6 = 60
10 x 7 = 70
10 x 8 = 80
10 x 9 = 90
10 x 10 = 100
```

The taskbar at the bottom shows the file path 'PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>' and various system icons.

19. Print the numbers between 100 to 200 which are completely divisible by 3 and 5.

```
#include <stdio.h>
```

```
int main() {
    printf("Numbers between 100 and 200 that are divisible by 3 and 5:\n");
    for (int i = 100; i <= 200; i++) {
        if (i % 3 == 0 && i % 5 == 0) {
            printf("%d\n", i);
        }
    }
    return 0;
}
```

A screenshot of the Visual Studio Code interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar shows various files: Calculation.java, call.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, C problem1.c (which is the active tab), Car.java, and Student.java. The main code editor window contains the following C code:

```
#include <stdio.h>
int main() {
    printf("Numbers between 100 and 200 that are divisible by 3 and 5:\n");
    for (int i = 100; i <= 200; i++) {
        if (i % 3 == 0 && i % 5 == 0) {
            printf("%d\n", i);
        }
    }
    return 0;
}
```

The terminal below shows the output of the code execution:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Numbers between 100 and 200 that are divisible by 3 and 5:
105
120
135
150
165
180
195
```

The taskbar at the bottom shows the Windows Start button, a search bar, and several pinned icons including File Explorer, Task View, Mail, Photos, OneDrive, Edge, Google Chrome, and File Explorer again. The system tray shows the date and time as 11/15/2023.

20. Find summation and average of all the numbers which are completely divisible by 3, 5 and 12 between 10 – 500. [Like 60 is divisible by all of those].

```
#include <stdio.h>
```

```
int main() {
    int sum = 0, count = 0;
```

```
    for (int i = 10; i <= 500; i++) {
        if (i % 3 == 0 && i % 5 == 0 && i % 12 == 0) {
```

```
            sum += i;
```

```
            count++;
```

```
}
```

```
}
```

```
double average = (double) sum / count;
```

```

printf("Sum of numbers: %d\n", sum);

printf("Average of numbers: %.2f\n", average);

return 0;
}

```

```

#include <stdio.h>
int main() {
    int sum = 0, count = 0;
    for (int i = 10; i <= 500; i++) {
        if (i % 3 == 0 && i % 5 == 0 && i % 12 == 0) {
            sum += i;
            count++;
        }
    }
    double average = (double) sum / count;
}

```

PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if (?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { ./tempCodeRunnerFile }

Sum of numbers: 2160

Average of numbers: 270.00

PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>

21. Suppose you are going to apply for admission in a college. If you got GPA 5 in SSC exam then you can apply for that college. Now give your GPA as input from the keyboard and print "YES" if you can apply otherwise print "NO".

```
#include <stdio.h>
```

```
int main() {
```

```
    float gpa;
```

```
    printf("Enter your GPA: ");
```

```
    scanf("%f", &gpa);
```

```

if (gpa == 5.0) {

    printf("YES, you can apply for admission.\n");

} else {

    printf("NO, you cannot apply for admission.\n");

}

return 0;
}

```

```

File Edit Selection View Go Run Terminal Help ← → A java practise
J Calculation.java J call1.java J Mycalculation.java J Vehicle.java J problem2.java JS test.js D test.html C problem1.c X J Car.java J Student.java D v ⚙ ...
oop > Capston > C problem1.c > ...
1 #include <stdio.h>
2
3 int main() {
4     float gpa;
5
6     printf("Enter your GPA: ");
7     scanf("%f", &gpa);
8
9     if (gpa == 5.0) {
10         printf("YES, you can apply for admission.\n");
11     } else {
12         printf("NO, you cannot apply for admission.\n");
13     }
14 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter your GPA: 3.75
NO, you cannot apply for admission.
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>

```

22. You are going to open a bank account . If your age is greater than 18 then you can open an account. Get your age by input and print "Yes" if you can open an account otherwise print "No".

```
#include <stdio.h>
```

```

int main() {

int age;

```

```
printf("Enter your age: ");
scanf("%d", &age);

if (age > 18) {
    printf("Yes, you can open a bank account.\n");
} else {
    printf("No, you cannot open a bank account.\n");
}

return 0;
}
```

23. Write a program to display "A" to "Z" using loop .

```
#include <stdio.h>
```

```
int main() {  
    printf("Alphabets from A to Z:\n");  
  
    for (char ch = 'A'; ch <= 'Z'; ch++) {  
        printf("%c ", ch);  
  
    }  
  
    printf("\n");  
  
    return 0;  
}
```



A screenshot of the Visual Studio Code (VS Code) interface. The top navigation bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar features icons for file types like Java (.java), C/C++ (.c), JavaScript (.js), and HTML (.html). The main editor window displays a C program named 'problem1.c' with the following code:

```
#include <stdio.h>
int main() {
    printf("Alphabets from A to Z:\n");
    for (char ch = 'A'; ch <= 'Z'; ch++) {
        printf("%c ", ch);
    }
    printf("\n");
    return 0;
}
```

The terminal tab at the bottom shows the execution of the program in a Windows command prompt (PS C:\...). The user runs 'cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston"' followed by 'gcc tempCodeRunnerFile.c -o tempCodeRunnerFile'. The output shows the program's output: 'Alphabets from A to Z:' followed by the sequence of letters from A to Z.

24. Write a program to produce the following output using loop
1 2 4 3 6 9 4 8 12 16 5 10 15 20 25 6 12 18 24
30 36

```
#include <stdio.h>
```

```
int main() {
```

int rows;

```
printf("Enter the number of rows: ");
```

```
scanf("%d", &rows);
```

```
for (int i = 1; i <= rows; i++) {
```

```
    int value = i;
```

```
    for (int j = 1; j <= i; j++) {
```

```
        printf("%d ", value);
```

```
        value += i;
```

```
}
```

```
printf("\n");
```

```
}
```

```
return 0;
```

```
}
```

The screenshot shows a code editor interface with a dark theme. The file tab bar at the top has several tabs, with 'problem1.c' being the active tab. The code area displays a C program that calculates the sum of integers from 1 to a given number of rows. The terminal below the editor shows the command-line output of running the program, which asks for the number of rows and then prints the sum of each row.

```
#include <stdio.h>
int main() {
    int rows;
    printf("Enter the number of rows: ");
    scanf("%d", &rows);
    for (int i = 1; i <= rows; i++) {
        int value = i;
        for (int j = 1; j <= i; j++) {
            printf("%d ", value);
            value += i;
        }
    }
}
```

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if (?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter the number of rows: 6
1
2 4
3 6 9
4 8 12 16
5 10 15 20 25
6 12 18 24 30 36
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

25..Write a program which will display all the prime numbers between 0 to N(N will be given by the user). If user gives N = 20, your code will print the following output
2 3 5 7 11 13 17 19

```
#include <stdio.h>

int main() {
    int rows;

    printf("Enter the number of rows: ");
    scanf("%d", &rows);

    for (int i = 1; i <= rows; i++) {
        int value = i;
        for (int j = 1; j <= i; j++) {
            printf("%d ", value);
            value += i;
        }
        printf("\n");
    }

    return 0;
}
```

A screenshot of the Visual Studio Code interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar for 'A java practise'. The left sidebar shows several open files: Calculation.java, call.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (which is the active file), Car.java, and Student.java. The main editor area contains the following C code:

```
3 int main() {
4     int rows;
5
6     printf("Enter the number of rows: ");
7     scanf("%d", &rows);
8
9     for (int i = 1; i <= rows; i++) {
10        int value = i;
11        for (int j = 1; j <= i; j++) {
12            printf("%d ", value);
13            value += i;
14        }
15        printf("\n");
16    }
17 }
```

The terminal below shows the execution of the program:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter the number of rows: 6
1
2 4
3 6 9
4 8 12 16
5 10 15 20 25
6 12 18 24 30 36
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The status bar at the bottom displays various system and application icons.

1. Write a program to print the following output: Hello world!!!

```
#include <stdio.h>
```

```
int main() {

    printf("Hello world!!!\n");

    return 0;

}
```

A screenshot of the Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. The title bar shows "A java practise". The left sidebar has icons for file operations like Open, Save, Find, and others. The main area displays a C code editor with the following content:

```
#include <stdio.h>
int main() {
    printf("Hello world!!!\n");
    return 0;
}
```

The terminal below shows the command line output:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Hello world!!!
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The bottom status bar shows file paths, line numbers, and other system information.

2. Write a code to print the following output: Name: Mr. X Univ: University of Dhaka

```
#include <stdio.h>
```

```
int main() {
    printf("Name: Mr. X Univ: University of Dhaka\n");
    return 0;
}
```

```
#include <stdio.h>
int main() {
    printf("Name: Mr. X Univ: University of Dhaka\n");
    return 0;
}
```

PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if (\$?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Name: Mr. X Univ: University of Dhaka
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>

3. Write a program to print the following output: * *** ***** ***** * **** *

```
#include <stdio.h>
```

```
int main() {
    int rows = 5;

    for (int i = 1; i <= rows; i++) {
        for (int j = 1; j <= i * 2 - 1; j++) {
            printf("*");
        }
        printf("\n");
    }

    return 0;
}
```

A screenshot of the Visual Studio Code interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar shows file icons for Calculation.java, call.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (which is the active tab), Car.java, and Student.java. The main code editor window displays the following C code:

```
oop > Capston > C problem1.c ...
3 int main() {
4     int rows = 5;
5
6     for (int i = 1; i <= rows; i++) {
7         for (int j = 1; j <= i * 2 - 1; j++) {
8             printf("*");
9         }
10        printf("\n");
11    }
12
13    return 0;
14 }
```

The terminal below shows the command line output:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
*
***  
*****  
*****  
*****  
*****  
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The bottom status bar shows various system and application icons.

4. Take three integer variables a, b and c. Assign values to the variables- a = 10, b = 20 and c = 30; Now print the sum of these three variables

```
#include <stdio.h>
```

```
int main() {
    int a = 10, b = 20, c = 30;
    int sum = a + b + c;
}
```

```
printf("Sum of a, b, and c: %d\n", sum);
```

```
return 0;
```

```
}
```

```
#include <stdio.h>
int main() {
    int a = 10, b = 20, c = 30;
    int sum = a + b + c;
    printf("Sum of a, b, and c: %d\n", sum);
    return 0;
}
```

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Sum of a, b, and c: 60
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

5. Read 2 integer values and store them in variables, named A and B and make the sum of these two variables, assigning its result to the variable X. Print X as shown below. The input file contains 2 integer values. A=10, B=9.

```
#include <stdio.h>
```

```
int main() {
    int A, B, X;
    printf("Enter values for A and B: ");
    scanf("%d %d", &A, &B);
```

```
X = A + B;
```

```
printf("X = %d\n", X);
```

```
return 0;
```

}

```
#include <stdio.h>
int main() {
    int A, B, X;
    printf("Enter values for A and B: ");
    scanf("%d %d", &A, &B);
    X = A + B;
    printf("X = %d\n", X);
    return 0;
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter values for A and B: 2 3
X = 5
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

6. The formula to calculate the area of a circumference is defined as $A = \pi \cdot R^2$. Considering to this problem that $\pi = 3.14159$:

```
#include <stdio.h>
```

```
int main() {
```

double R = 2.0;

double A;

$$A = 3.14159 * R * R;$$

```
printf("Area of the circumference: %.4lf\n", A);
```

```
return 0;
```

}

A screenshot of the Visual Studio Code interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar shows several open files: Calculation.java, call.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (which is the active tab), Car.java, and Student.java. The main code editor window displays the following C code:

```
#include <stdio.h>
int main() {
    double R = 2.0;
    double A;
    A = 3.14159 * R * R;
    printf("Area of the circumference: %.4lf\n", A);
    return 0;
}
```

The terminal below the editor shows the command-line output of running the program:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Area of the circumference: 12.5664
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

7. Read two floating points' values of double precision A and B, corresponding to two student's grades. After this, calculate the student's average, considering that grade A has weight 3.5 and B has weight 7.5. Each grade can be from zero to ten, always with one digit after the decimal point.

```
#include <stdio.h>
```

```
int main() {
    double A, B, average;
    printf("Enter grades A and B: ");
    scanf("%lf %lf", &A, &B);
```

```
    average = (A * 3.5 + B * 7.5) / (3.5 + 7.5);
```

```
    printf("Average: %.5lf\n", average);
```

```
return 0;
```

}

A screenshot of a terminal window in a dark-themed code editor. The window title is "A java practise". The terminal shows the following command and output:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter grades A and B: 2 3
Average: 2.68182
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The terminal interface includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS. The status bar at the bottom shows file paths, line counts, and other system information.

8. Read three values (variables A=5.0, B=6.0 and C=7.0), which are the three student's grades. Then, calculate the average, considering that grade A has weight 2, grade B has weight 3 and the grade C has weight 5. Consider that each grade can go from 0 to 10.0, always with one decimal place. Output: Average=6.3

```
#include <stdio.h>
```

```
int main() {
    double A = 5.0, B = 6.0, C = 7.0;
    double average;
    average = (A * 2 + B * 3 + C * 5) / (2 + 3 + 5);
    printf("Average: %.1f\n", average);
    return 0;
}
```

```
}
```

```
#include <stdio.h>
int main() {
    double A = 5.0, B = 6.0, C = 7.0;
    double average;
    average = (A * 2 + B * 3 + C * 5) / (2 + 3 + 5);
    printf("Average: %.1lf\n", average);
    return 0;
}
```

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter grades A and B: 2 3
Average: 2.68182
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

9. Read four integer values named A, B, C and D. Calculate and print the difference of product A and B by the product of C and D ($A * B - C * D$).

```
#include <stdio.h>
```

```
int main() {
    int A, B, C, D;
    printf("Enter values for A, B, C, and D: ");
    scanf("%d %d %d %d", &A, &B, &C, &D);

    int difference = A * B - C * D;

    printf("Difference of product A and B by the product of C and D: %d\n", difference);
```

```
return 0;
```

```
}
```

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar has icons for file types like Java (.java), JavaScript (.js), and C/C++ (.c). The main editor area contains a C program named 'problem1.c' with the following code:

```
#include <stdio.h>
int main() {
    int A, B, C, D;
    printf("Enter values for A, B, C, and D: ");
    scanf("%d %d %d %d", &A, &B, &C, &D);
    int difference = A * B - C * D;
    printf("Difference of product A and B by the product of C and D: %d\n", difference);
    return 0;
}
```

Below the editor is a terminal window showing the command-line output of running the program:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter values for A, B, C, and D: 2 3 6 8
Difference of product A and B by the product of C and D: -42
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The bottom status bar shows the current file path as 'C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\problem1.c', line 15, column 1 (303 selected), and other system information.

10. Write a program that reads an employee's number, his/her worked hours number in a month and the amount he received per hour. Print the employee's number and salary that he/she will receive at end of the month, with two decimal places. Input Samples: 25 100 5.50 Output: NUMBER = 25 SALARY = U\$ 550.00

```
#include <stdio.h>
```

```
int main() {
    int employeeNumber;
    double hoursWorked, hourlyRate, salary;

    printf("Enter employee number, hours worked, and hourly rate: ");
    scanf("%d %lf %lf", &employeeNumber, &hoursWorked, &hourlyRate);

    salary = hoursWorked * hourlyRate;
```

```
printf("NUMBER = %d SALARY = U$ %.2lf\n", employeeNumber, salary);

return 0;

}
```

A screenshot of the Visual Studio Code (VS Code) interface. The top navigation bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar containing "A java practise". The left sidebar features icons for file operations like Open, Save, Find, and Settings. The main workspace shows a Java file named "problem1.c" with the following code:

```
3 int main() {
4     int employeeNumber;
5     double hoursWorked, hourlyRate, salary;
6
7     printf("Enter employee number, hours worked, and hourly rate: ");
8     scanf("%d %lf %lf", &employeeNumber, &hoursWorked, &hourlyRate);
9
10    salary = hoursWorked * hourlyRate;
11
12    printf("NUMBER = %d SALARY = U$ %.2lf\n", employeeNumber, salary);
13
14    return 0;
15 }
```

The bottom navigation bar includes PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (which is selected), and PORTS. The terminal tab displays the following command-line session:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter employee number, hours worked, and hourly rate: 2 2.2 3.6
NUMBER = 2 SALARY = U$ 7.92
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The status bar at the bottom shows the current file is "problem1.c", and other details like line 16, column 1, and file encoding.

11. Take four double variables x , y , z and avg . Assign values to x , y and z as you want. Now calculate the average of x , y and z and assign it to avg . Finally print the average value like Average of x , y and z is: —

```
#include <stdio.h>
```

```
int main() {  
    double x, y, z, avg;  
  
    // Assign values to x, y, and z  
  
    x = 10.5;  
    y = 20.3;  
    z = 15.8;
```

```

// Calculate the average

avg = (x + y + z) / 3;

// Print the average

printf("Average of x, y, and z is: %.2lf\n", avg);

return 0;
}

```

```

File Edit Selection View Go Run Terminal Help ← → A java practise
J Calculation.java J cal1.java J Mycalculation.java J Vehicle.java J problem2.java JS test.js O test.html C problem1.c X J Car.java J Student.java D v ⚙ ...
oop > Capston > C problem1.c ...
1 #include <stdio.h>
2
3 int main() {
4     double x, y, z, avg;
5
6     // Assign values to x, y, and z
7     x = 10.5;
8     y = 20.3;
9     z = -15.8;
10
11     // Calculate the average
12     avg = (x + y + z) / 3;
13
14     // Print the average
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Code + ▾ □ ⌂ ... ×
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { ./tempCodeRunnerFile }
Average of x, y, and z is: 15.53
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>

```

The screenshot shows a code editor with a dark theme. The file tab bar at the top has several tabs, with 'problem1.c' currently active. The code area contains a C program that calculates the average of three variables (x, y, z) and prints the result. Below the code editor is a terminal window showing the command to compile and run the program, followed by its output. The system tray at the bottom indicates it's 11:10 PM on 11/15/2023.

12. Print "Bangladesh" without quotation sign by using ascii values. [Use the given ascii values B=66, a=97, n=110, g=103, l=108, a=97, d=100, e=101, s=115, h=104.]

```
#include <stdio.h>
```

```

int main() {

printf("%c%c%c%c%c%c%c%c\n", 66, 97, 110, 103, 108, 97, 100, 101, 115, 104);
}

```

```
    return 0;
```

```
}
```

```
A java practise
oop > Capston > C problem1.c ...
1 #include <stdio.h>
2
3 int main() {
4     printf("%c%c%c%c%c%c%c%c\n", 66, 97, 110, 103, 108, 97, 100, 101, 115, 104);
5
6     return 0;
7 }
8

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Bangladesh
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>

Ln 8. Col 1 Spaces: 4 UTF-8 CRLF ⚡ Go Live ⚡ Port : 5500 Win32 ⚡ Go Live ⚡
25°C Haze 11:11 PM 11/15/2023
```

13. Make a program that reads a seller's name, his/her fixed salary and the sale's total made by himself/herself in the month (in money). Considering that this seller receives 15% over all products sold, write the final salary (total) of this seller at the end of the month, with two decimal places. Input Samples: JOAO 500.00 1230.30
Output Samples: TOTAL = R\$ 684.54

```
#include <stdio.h>
```

```
int main() {
    char sellerName[50];
    double fixedSalary, totalSales, finalSalary;

    printf("Enter seller's name, fixed salary, and total sales: ");
    scanf("%s %lf %lf", sellerName, &fixedSalary, &totalSales);
```

```
// Calculate final salary (including 15% of total sales)  
  
finalSalary = fixedSalary + 0.15 * totalSales;  
  
printf("TOTAL = R$ %.2lf\n", finalSalary);  
  
return 0;  
}
```

A screenshot of the Visual Studio Code interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar. The left sidebar has icons for file types like Java, C, JavaScript, and HTML. The main editor window shows a C program named problem1.c. The code prompts the user for a seller's name, fixed salary, and total sales, then calculates a 15% bonus and prints the total salary. The terminal below shows the command to run the program and its output.

```
#include <stdio.h>
int main() {
    char sellerName[50];
    double fixedSalary, totalSales, finalSalary;
    printf("Enter seller's name, fixed salary, and total sales: ");
    scanf("%s %lf %lf", sellerName, &fixedSalary, &totalSales);
    finalSalary = fixedSalary + 0.15 * totalSales;
    printf("TOTAL = R$ %.2lf\n", finalSalary);
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter seller's name, fixed salary, and total sales: 2 3.5 2.2
TOTAL = R$ 3.83
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

14. Rahul got 65.00 on physics, 83.50 on mathematics, 85.75 on C programming and 67.50 on English. Now write a program to calculate the average of his marks on 4 subjects and print it up to 2 digits after the decimal point. [The result should look like: XX.XX]

```
#include <stdio.h>
```

```
int main() {
```

```
double physics = 65.00, mathematics = 83.50, cProgramming = 85.75, english = 67.50;  
double average;
```

```
// Calculate the average  
  
average = (physics + mathematics + cProgramming + english) / 4;  
  
// Print the average  
  
printf("Average of marks: %.2lf\n", average);  
  
return 0;  
}
```

A screenshot of the Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar features icons for file operations like Open, Save, Find, and Settings. The main editor area displays a C program named 'problem1.c' with the following code:

```
1 #include <stdio.h>
2
3 int main() {
4     double physics = 65.00, mathematics = 83.50, cProgramming = 85.75, english = 67.50;
5     double average;
6
7     // Calculate the average
8     average = (physics + mathematics + cProgramming + english) / 4;
9
10    // Print the average
11    printf("Average of marks: %.2lf\n", average);
12
13    return 0;
14 }
```

The bottom terminal tab shows the command-line output of running the program:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Average of marks: 75.44
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

15. In this problem, the task is to read a code of a product 1, the number of units of product 1, the price for one unit of product 1, the code of a product 2, the number of units of product 2 and the price for one unit of product 2. After this, calculate and show the amount to be paid. Input Samples: 12 1 5.30 16 2 5.10 Output Sample: VALOR A PAGAR: R\$ 15.50

```
#include <stdio.h>
```

```
int main() {  
    int code1, units1, code2, units2;
```

```

double price1, price2, amount;

printf("Enter code, units, and price for product 1: ");
scanf("%d %d %lf", &code1, &units1, &price1);

printf("Enter code, units, and price for product 2: ");
scanf("%d %d %lf", &code2, &units2, &price2);

// Calculate the amount to be paid
amount = (units1 * price1) + (units2 * price2);

printf("VALOR A PAGAR: R$ %.2lf\n", amount);

return 0;
}

```

The screenshot shows the Microsoft Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar. The left sidebar has icons for files, folders, and other project-related items. The main editor area displays a C program with syntax highlighting for keywords like int, main, and operators like * and +. The code prompts the user to enter code, units, and price for two products, calculates the total amount, and prints it back. Below the editor is a terminal window showing the command-line interface where the program is run and its output is displayed. The bottom status bar shows file paths, line numbers, and other system information.

```

PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { ./tempCodeRunnerFile }
Enter code, units, and price for product 1: 12 22 33
Enter code, units, and price for product 2: 25 66 33
VALOR A PAGAR: R$ 2904.00
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>

```

16. Take two integer variables $i = 0$ and $j = 0$. Now write the output of the following program without running the code. 17. Take four integer variables a , b , x and y . Scan the values of the variables from user using `scanf()` function. Now print the output of the following equation: $(a * b) + (x * y)$

```
#include <stdio.h>

int main() {
    int a, b, x, y;

    // Scan values of variables
    printf("Enter values for a, b, x, and y: ");
    scanf("%d %d %d", &a, &b, &x, &y);

    // Calculate and print the output of the equation
    printf("Result of (a * b) + (x * y): %d\n", (a * b) + (x * y));

    return 0;
}
```

18. Take temperature of Dhaka city as input in Celsius scale from the user using `scanf()` function and convert it to Fahrenheit and print it.[Formula: $F = C(9/5) + 32$]

```
#include <stdio.h>

int main() {
    double celsius, fahrenheit;

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

```
// Convert to Fahrenheit  
  
fahrenheit = celsius * (9.0 / 5.0) + 32;  
  
// Print the result  
  
printf("Temperature in Fahrenheit: %.2lf\n", fahrenheit);  
  
return 0;  
}
```

The screenshot shows a terminal window with the following text:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter temperature in Celsius: 22
Temperature in Fahrenheit: 71.60
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

19. Take a small letter alphabet as input from the user and print the capital version of that letter. [If user gives input 'a' you should print 'A']

```
#include <stdio.h>
```

```
int main() {  
    char smallLetter, capitalLetter;
```

```
// Input a small letter alphabet  
  
printf("Enter a small letter alphabet: ");  
  
scanf(" %c", &smallLetter);  
  
// Convert to capital letter  
  
capitalLetter = smallLetter - 32;  
  
// Print the capital letter  
  
printf("Capital letter: %c\n", capitalLetter);  
  
return 0;  
}
```

A screenshot of the Visual Studio Code (VS Code) interface. The top bar shows the file menu (File, Edit, Selection, View, Go, Run, Terminal, Help), a back/forward navigation bar, and a search bar containing "A java practise". The left sidebar has icons for files, folders, and other workspace items. The main editor area contains a C program:

```
1 #include <stdio.h>
2
3 int main() {
4     char smallLetter, capitalLetter;
5
6     // Input a small letter alphabet
7     printf("Enter a small letter alphabet: ");
8     scanf(" %c", &smallLetter);
9
10    // Convert to capital letter
11    capitalLetter = smallLetter - 32;
12
13    // Print the capital letter
14    printf("Capital letter: %c\n", capitalLetter);
```

The status bar at the bottom shows the terminal command: PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if (\$?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (\$?) { .\tempCodeRunnerFile }. It also displays system information like battery level (24%), temperature (24°C), and network status (Haze).

20. Make a program that calculates and shows the volume of a sphere being provided the value of its radius (R). The formula to calculate the volume is: $(4/3) * \pi * R^3$. Consider (assign) for pi the value 3.14159. Tip: Use $(4/3.0)$ or $(4.0/3)$ in your formula, because some languages (including C++) assume that the division's result between two integers is another integer. Input sample: 3 Output Sample: VOLUME = 113.097

```
#include <stdio.h>
#include <math.h>

int main() {
    double radius, volume;

    // Input radius of the sphere
    printf("Enter the radius of the sphere: ");
    scanf("%lf", &radius);

    // Calculate the volume
    volume = (4.0 / 3.0) * M_PI * pow(radius, 3);

    // Print the volume
    printf("VOLUME = %.3lf\n", volume);

    return 0;
}
```

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar has icons for file types like Java (.java), JavaScript (.js), and HTML (.html). The main editor area contains a C program named 'problem1.c' with code to calculate the volume of a sphere based on user input for radius. Below the editor are tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS. The TERMINAL tab is active, displaying command-line output from a terminal window. The output shows the command 'cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston"' being run, followed by 'gcc tempCodeRunnerFile.c -o tempCodeRunnerFile', then 'Enter the radius of the sphere: 5', and finally 'VOLUME = 523.600'. The bottom status bar shows file paths, line numbers (Ln 5, Col 27), spaces (Spaces: 4), encoding (UTF-8), and other system information.

```
#include <stdio.h>

int main() {
    double radius, volume;
    // Input radius of the sphere
    printf("Enter the radius of the sphere: ");
    scanf("%lf", &radius);
    // Calculate the volume
    volume = (4.0 / 3.0) * 3.1416 * pow(radius, 3);
    // Print the volume
}
```

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston"
PS C:\Users\fahim\OneDrive\Documents\A java practise> gcc tempCodeRunnerFile.c -o tempCodeRunnerFile
PS C:\Users\fahim\OneDrive\Documents\A java practise> Enter the radius of the sphere: 5
PS C:\Users\fahim\OneDrive\Documents\A java practise> VOLUME = 523.600
PS C:\Users\fahim\OneDrive\Documents\A java practise>
```

21. Suppose you are going to apply for admission in a college. If you got GPA 5 in SSC exam then you can apply for that college. Now give your GPA as input from the keyboard and print "YES" if you can apply otherwise print "NO".

```
#include <stdio.h>
```

```
int main() {
    float gpa;
    printf("Enter your GPA: ");
    scanf("%f", &gpa);

    if (gpa == 5.0) {
        printf("YES, you can apply for admission.\n");
    } else {
        printf("NO, you cannot apply for admission.\n");
    }
}
```

```
return 0;
```

```
}
```

The screenshot shows a code editor interface with a dark theme. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. A search bar at the top right contains the text "A java practise". The left sidebar has icons for file operations like Open, Save, Find, and others. The main workspace displays a C program named "problem1.c". The code reads a floating-point value for GPA, checks if it's 5.0, and prints "YES" or "NO" accordingly. The terminal tab at the bottom shows the command-line output of running the program.

```
oop > Capston > C problem1.c ...
5
6     ... printf("Enter your GPA: ");
7     ... scanf("%f", &gpa);
8
9     if (gpa == 5.0) {
10        ... printf("YES, you can apply for admission.\n");
11    } else {
12        ... printf("NO, you cannot apply for admission.\n");
13    }
14
15    ... return 0;
16 }
17
```

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter your GPA: 3.4
NO, you cannot apply for admission.
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

22. Make a program that reads three floating point values: A, B and C. Then, calculate and show: a) the area of the rectangled triangle that has base A and height C. b) the area of the radius's circle C. ($\pi = 3.14159$) c) the area of the trapezium which has A and B by base, and C by height. d) the area of the square that has side B. e) the area of the rectangle that has sides A and B.

```
#include <stdio.h>
```

```
#include <math.h>
```

```
int main() {
```

```
    float A, B, C;
```

```
    float areaTriangle, areaCircle, areaTrapezium, areaSquare, areaRectangle;
```

```
    printf("Enter three floating point values A, B, and C: ");
```

```
    scanf("%f %f %f", &A, &B, &C);
```

```
// Calculate areas

areaTriangle = 0.5 * A * C;
areaCircle = M_PI * pow(C, 2);
areaTrapezium = 0.5 * (A + B) * C;
areaSquare = pow(B, 2);
areaRectangle = A * B;

// Print the areas

printf("a) Area of the rectangle triangle: %.2f\n", areaTriangle);
printf("b) Area of the circle: %.2f\n", areaCircle);
printf("c) Area of the trapezium: %.2f\n", areaTrapezium);
printf("d) Area of the square: %.2f\n", areaSquare);
printf("e) Area of the rectangle: %.2f\n", areaRectangle);

return 0;
```

```
}
```

```

1 #include <stdio.h>
2 #include <math.h>
3
4 int main() {
5     float A, B, C;
6     float areaTriangle, areaCircle, areaTrapezium, areaSquare, areaRectangle;
7
8     printf("Enter three floating point values A, B, and C: ");
9     scanf("%f %f %f", &A, &B, &C);
10
11     // Calculate areas
12     areaTriangle = 0.5 * A * C;
13     areaCircle = 3.1416 * pow(C, 2);
14     areaTrapezium = 0.5 * (A + B) * C;

```

PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if (\$?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (\$?) { ./tempCodeRunnerFile }

Enter three floating point values A, B, and C: 5 6 3

a) Area of the rectangle triangle: 7.50
b) Area of the circle: 28.27
c) Area of the trapezium: 16.50
d) Area of the square: 36.00
e) Area of the rectangle: 30.00

PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>

23. Take a value from user and assume that- it is the number of his math exam. Now you have to write a program which shows the grade depending on the given scale. (A+: 80-100, A: 70-79, A- : 60-69, B: 50-59, C: 40-49, D :33-39 ,F :0-32)

```
#include <stdio.h>
```

```

int main() {

    int mathExam;

    printf("Enter your math exam score: ");

    scanf("%d", &mathExam);

    if (mathExam >= 80 && mathExam <= 100) {

        printf("Grade: A+\n");

    } else if (mathExam >= 70 && mathExam < 80) {

        printf("Grade: A\n");

    } else if (mathExam >= 60 && mathExam < 70) {


```

```
printf("Grade: A\n");
} else if (mathExam >= 50 && mathExam < 60) {
    printf("Grade: B\n");
} else if (mathExam >= 40 && mathExam < 50) {
    printf("Grade: C\n");
} else if (mathExam >= 33 && mathExam < 40) {
    printf("Grade: D\n");
} else if (mathExam >= 0 && mathExam < 33) {
    printf("Grade: F\n");
} else {
    printf("Invalid input\n");
}
return 0;
}
```

A screenshot of the Visual Studio Code (VS Code) interface. The top bar shows the menu: File, Edit, Selection, View, Go, Run, Terminal, Help. The title bar says "A java practise". The left sidebar has icons for file operations like Open, Save, Find, and others. The main editor area contains Java code for calculating grades based on exam scores:

```
5
6     printf("Enter your math exam score: ");
7     scanf("%d", &mathExam);
8
9     if (mathExam >= 80 && mathExam <= 100) {
10        printf("Grade: A+\n");
11    } else if (mathExam >= 70 && mathExam < 80) {
12        printf("Grade: A\n");
13    } else if (mathExam >= 60 && mathExam < 70) {
14        printf("Grade: A-\n");
15    } else if (mathExam >= 50 && mathExam < 60) {
16        printf("Grade: B\n");
17    } else if (mathExam >= 40 && mathExam < 50) {
18        printf("Grade: C\n");
19    } else {
20        printf("Grade: D\n");
21    }
22 }
```

The bottom navigation bar includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS. The TERMINAL tab is active, showing command-line output:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter your math exam score: 6.2
Grade: F
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

24. Calculate a car's average consumption being provided the total distance traveled (in Km) and the spent fuel total (in liters). Input Sample: 500 35.0 Output Sample: 14.286 km/l

```
#include <stdio.h>
```

```
int main() {
```

```
    float distance, fuel;
```

```
    float averageConsumption;
```

```
    printf("Enter total distance traveled (in Km) and spent fuel total (in liters): ");
```

```
    scanf("%f %f", &distance, &fuel);
```

```
// Calculate average consumption
```

```
    averageConsumption = distance / fuel;
```

```
    printf("Average consumption: %.3f km/l\n", averageConsumption);
```

```
    return 0;
```

```
}
```

A screenshot of the Visual Studio Code interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar shows several open files: Calculation.java, call.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (the active tab), Car.java, and Student.java. The main editor area contains the following C code:

```
1 #include <stdio.h>
2
3 int main() {
4     float distance, fuel;
5     float averageConsumption;
6
7     printf("Enter total distance traveled (in Km) and spent fuel total (in liters): ");
8     scanf("%f %f", &distance, &fuel);
9
10    // Calculate average consumption
11    averageConsumption = distance / fuel;
12
13    printf("Average consumption: %.3f km/l\n", averageConsumption);
14 }
```

The terminal below shows the command to run the code and its output:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter total distance traveled (in Km) and spent fuel total (in liters): 5 6
Average consumption: 0.833 km/l
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The bottom status bar shows file paths, line numbers, and other system information.

25. You are going to open a bank account. If your age is greater than 18 then you can open an account. Get your age by input and print "Yes" if you can open an account otherwise print "No".

```
#include <stdio.h>
```

```
int main() {
```

```
    int age;
```

```
    printf("Enter your age: ");
```

```
    scanf("%d", &age);
```

```
    if (age > 18) {
```

```
        printf("Yes, you can open a bank account.\n");
```

```
    } else {
```

```
        printf("No, you cannot open a bank account.\n");
```

```
}
```

```
return 0;
```

```
}
```

A screenshot of a terminal window titled "A java practise". The window shows a C program named "problem1.c" with code to determine if a user can open a bank account based on their age. The terminal output shows the program running and prompting for an age, with the result "No, you cannot open a bank account." displayed.

```
#include <stdio.h>
int main() {
    int age;
    printf("Enter your age: ");
    scanf("%d", &age);
    if (age > 18) {
        printf("Yes, you can open a bank account.\n");
    } else {
        printf("No, you cannot open a bank account.\n");
    }
}
```

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter your age: 17
No, you cannot open a bank account.
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

26. Write a program which determines whether a number is ODD or EVEN.

```
#include <stdio.h>
```

```
int main() {
```

```
    int number;
```

```
    printf("Enter an integer: ");
```

```
    scanf("%d", &number);
```

```
    if (number % 2 == 0) {
```

```
        printf("EVEN\n");
```

```
    } else {
```

```
    printf("ODD\n");  
}
```

```
return 0;
```

```
}
```

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar has icons for file types like Java, JavaScript, and C. The main editor tab is 'problem1.c'. The code in the editor is:

```
1 #include <stdio.h>  
2  
3 int main() {  
4     int age;  
5  
6     printf("Enter your age: ");  
7     scanf("%d", &age);  
8  
9     if (age > 18) {  
10         printf("Yes, you can open a bank account.\n");  
11     } else {  
12         printf("No, you cannot open a bank account.\n");  
13     }  
14 }
```

The terminal below the editor shows the command line output:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if (?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }  
Enter your age: 17  
No, you cannot open a bank account.  
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The bottom status bar shows system information like battery level (24%), weather (Haze), and system time (11:29 PM, 11/15/2023).

27. Take an integer number as input from user and print "Yes" if the number is divisible by 3 and 5. And print "No" if the number is not.

```
#include <stdio.h>
```

```
int main() {  
  
    int number;  
  
    printf("Enter an integer: ");  
  
    scanf("%d", &number);
```

```

if (number % 3 == 0 && number % 5 == 0) {

    printf("Yes\n");

} else {

    printf("No\n");

}

return 0;
}

```

```

File Edit Selection View Go Run Terminal Help ← → A java practise
Calculation.java cal1.java Mycalculation.java Vehicle.java problem2.java test.js test.html c problem1.c Car.java Student.java
1 #include <stdio.h>
2
3 int main() {
4     int number;
5
6     printf("Enter an integer: ");
7     scanf("%d", &number);
8
9     if (number % 3 == 0 && number % 5 == 0) {
10         printf("Yes\n");
11     } else {
12         printf("No\n");
13     }
14 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if (?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter an integer: 3
No
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston> □

Ln 8, Col 1 Spaces: 4 UTF-8 CRLF ⌂ C ⌂ Go Live ⌂ Port: 5500 Win32 ⌂ Go Live ⌂ 11:29 PM 11/15/2023
24% Haze

```

28. Read the four values corresponding to the x and y axes of two points in the plane, p1 (x_1, y_1) and p2 (x_2, y_2) and calculate the distance between them, showing four decimal places after the comma, according to the formula: Distance =

```
#include <stdio.h>
```

```
#include <math.h>
```

```
int main() {
```

```
    float x1, y1, x2, y2;
```

float distance;

```
printf("Enter coordinates (x1, y1) and (x2, y2): ");

scanf("%f %f %f %f", &x1, &y1, &x2, &y2);

// Calculate distance

distance = sqrt(pow((x2 - x1), 2) + pow((y2 - y1), 2));

// Print the distance

printf("Distance: %.4f\n", distance);

return 0;

}
```

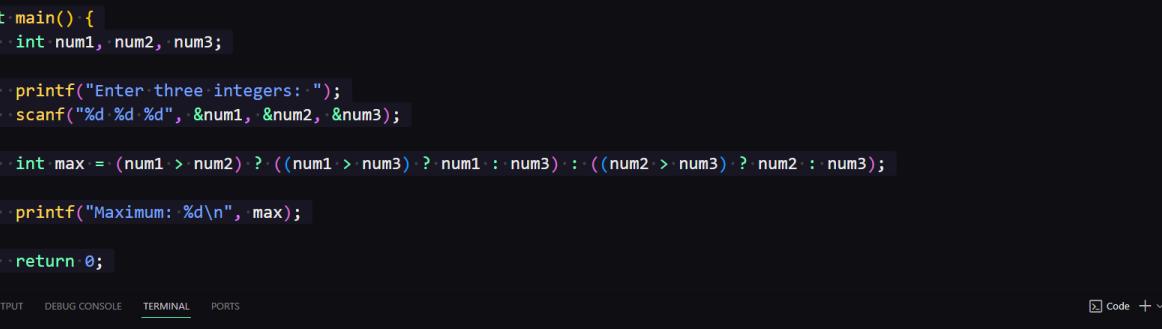
The screenshot shows a terminal window with the following content:

```
File Edit Selection View Go Run Terminal Help ↵ → A Java practise
Calculation.java call.java Mycalculation.java Vehicle.java problem2.java test.js test.html C problem1.c Car.java Student.java
oop > Capston > C problem1.c > main()
1 #include <stdio.h>
2 #include <math.h>
3
4 int main() {
5     float x1, y1, x2, y2;
6     float distance;
7
8     printf("Enter coordinates (x1, y1) and (x2, y2): ");
9     scanf("%f %f %f %f", &x1, &y1, &x2, &y2);
10
11     // Calculate distance
12     distance = sqrt(pow((x2 - x1), 2) + pow((y2 - y1), 2));
13
14     // Print the distance
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if (?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter coordinates (x1, y1) and (x2, y2): 2 3 9.2 6.2
Distance: 7.8791
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

29. Take 3 integers from user using scanf() function and write a program to find the maximum one.

```
#include <stdio.h>
```

```
int main() {  
    int num1, num2, num3;  
  
    printf("Enter three integers: ");  
    scanf("%d %d %d", &num1, &num2, &num3);  
  
    int max = (num1 > num2) ? ((num1 > num3) ? num1 : num3) : ((num2 > num3) ? num2 : num3);  
  
    printf("Maximum: %d\n", max);  
  
    return 0;  
}
```



```
#include <stdio.h>
int main() {
    int num1, num2, num3;
    printf("Enter three integers: ");
    scanf("%d %d %d", &num1, &num2, &num3);
    int max = (num1 > num2) ? ((num1 > num3) ? num1 : num3) : ((num2 > num3) ? num2 : num3);
    printf("Maximum: %d\n", max);
    return 0;
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter three integers: 5 6 9
Maximum: 9
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

30. Two cars (X and Y) leave in the same direction. The car X leaves with a constant speed of 60 km/h and the car Y leaves with a constant speed of 90 km / h. In one hour (60 minutes) the car Y can get a distance of 30 kilometers from the X car, in other words, it can get away one kilometer for each 2 minutes. Read the distance

(in km) and calculate how long it takes (in minutes) for the car Y to take this distance in relation to the other car.

```
#include <stdio.h>
```

```
int main() {  
    int distance;  
    int speedX = 60, speedY = 90;  
    int time;  
  
    printf("Enter the distance between the cars (in km): ");  
    scanf("%d", &distance);  
  
    // Calculate time  
    time = (distance * 60) / (speedY - speedX);  
  
    printf("Time taken for car Y to cover the distance: %d minutes\n", time);  
  
    return 0;  
}
```

The screenshot shows a code editor with a dark theme. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar has icons for file operations like Open, Save, Find, and Copy/Paste. The main editor area contains a C program:

```
#include <stdio.h>
int main() {
    int distance;
    int speedX = 60, speedY = 90;
    int time;
    printf("Enter the distance between the cars (in km): ");
    scanf("%d", &distance);
    time = (distance * 60) / (speedY - speedX);
    printf("Time taken for car Y to cover the distance: %d minutes\n", time);
}
```

The terminal below the editor shows the command line and its output:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter the distance between the cars (in km): 6
Time taken for car Y to cover the distance: 12 minutes
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The bottom status bar shows system information like battery level, temperature (24°C Haze), and system date/time (11/15/2023 11:32 PM).

31. Take a small letter alphabet as input and print whether it is VOWEL or CONSONANT.

```
#include <stdio.h>
```

```
int main() {
```

```
    char alphabet;
```

```
    printf("Enter a small letter alphabet: ");
```

```
    scanf(" %c", &alphabet);
```

```
    switch (alphabet) {
```

```
        case 'a':
```

```
        case 'e':
```

```
        case 'i':
```

```
        case 'o':
```

```
case 'u':  
    printf("VOWEL\n");  
  
    break;  
  
default:  
    printf("CONSONANT\n");  
  
}  
  
return 0;  
}
```

A screenshot of the Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A Java practise'. The left sidebar contains icons for file operations like Open, Save, Find, and others. The main editor area shows a C program named 'main.c' with the following code:

```
1 #include <stdio.h>
2
3 int main() {
4     char alphabet;
5
6     printf("Enter a small letter alphabet: ");
7     scanf(" %c", &alphabet);
8
9     switch (alphabet) {
10         case 'a':
11         case 'e':
12         case 'i':
13         case 'o':
14         case 'u':
```

The status bar at the bottom displays the terminal command: PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if (\$?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (\$?) { .\tempCodeRunnerFile }. It also shows the current file path as PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>. The bottom right corner shows system information: 11:33 PM, 11/15/2023, and various browser and system icons.

32. Little John wants to calculate and show the amount of spent fuel liters on a trip, using a car that does 12 Km/L. For this, he would like you to help him through a simple program. To perform the calculation, you have to read spent time (in hours) and the same average speed (km/h). In this way, you can get distance and then, calculate how many liters would be needed. Show with three decimal places after the point. input Sample: 10 85 Output: 70.833

```
#include <stdio.h>
```

```
int main() {
    float spentTime, averageSpeed;
    float distance, fuelNeeded;

    printf("Enter spent time (in hours) and average speed (km/h): ");
    scanf("%f %f", &spentTime, &averageSpeed);

    // Calculate distance
    distance = spentTime * averageSpeed;

    // Calculate fuel needed
    fuelNeeded = distance / 12;

    // Print the result
    printf("%.3f\n", fuelNeeded);

    return 0;
}
```

```
#include <stdio.h>
int main() {
    float spentTime, averageSpeed;
    float distance, fuelNeeded;
    printf("Enter spent time (in hours) and average speed (km/h): ");
    scanf("%f %f", &spentTime, &averageSpeed);
    // Calculate distance
    distance = spentTime * averageSpeed;
    // Calculate fuel needed
    fuelNeeded = distance / 12;
}
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston"
PS C:\Users\fahim\OneDrive\Documents\A java practise> if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile }
PS C:\Users\fahim\OneDrive\Documents\A java practise> .\tempCodeRunnerFile
Enter spent time (in hours) and average speed (km/h): 5.2 6.3
2.730
PS C:\Users\fahim\OneDrive\Documents\A java practise>
```

33. Write a program that takes an integer as input [1 – 12] and print the corresponding month name. [If user gives input '1' you should print 'January'].

```
#include <stdio.h>
```

```
int main() {
    int month;
    printf("Enter a number between 1 and 12: ");
    scanf("%d", &month);
```

```
switch (month) {
```

```
    case 1:
```

```
        printf("January\n");
```

```
        break;
```

```
    case 2:
```

```
printf("February\n");
```

```
break;
```

```
case 3:
```

```
printf("March\n");
```

```
break;
```

```
case 4:
```

```
printf("April\n");
```

```
break;
```

```
case 5:
```

```
printf("May\n");
```

```
break;
```

```
case 6:
```

```
printf("June\n");
```

```
break;
```

```
case 7:
```

```
printf("July\n");
```

```
break;
```

```
case 8:
```

```
printf("August\n");
```

```
break;
```

```
case 9:
```

```
printf("September\n");
```

```
break;
```

```
case 10:
```

```
printf("October\n");
```

```
break;
```

```
case 11:
```

```
printf("November\n");  
  
break;  
  
case 12:  
  
printf("December\n");  
  
break;  
  
default:  
  
printf("Invalid input\n");  
  
}
```

```
return 0;
```

}



A screenshot of the Visual Studio Code (VS Code) interface. The top navigation bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar. The left sidebar contains icons for file operations like Open, Save, Find, and others. The main editor area shows a C program with code for printing months from June to September. The terminal at the bottom shows the command to compile and run the program, followed by the output 'June'.

```
24     break;
25 case 6:
26     printf("June\n");
27     break;
28 case 7:
29     printf("July\n");
30     break;
31 case 8:
32     printf("August\n");
33     break;
34 case 9:
35     printf("September\n");
36     break;
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter a number between 1 and 12: 6
June
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

34. You are given the rank and salary scale of a company
Rank – salary 1 – 2,50,000 BDT 2 – 2,10,000 BDT 3 – 1,50,000 BDT 4 – 80,000 BDT >=5 – 50,000 BDT Now, take the rank as input from the user and print the salary of the given rank. [If user gives input '3' you should print 'Your Salary: 1,50,000 BDT']

```
#include <stdio.h>
```

```
int main() {
    int rank;

    printf("Enter your rank: ");
    scanf("%d", &rank);

    switch (rank) {
        case 1:
            printf("Your Salary: 2,50,000 BDT\n");
            break;
        case 2:
            printf("Your Salary: 2,10,000 BDT\n");
            break;
        case 3:
            printf("Your Salary: 1,50,000 BDT\n");
            break;
        case 4:
            printf("Your Salary: 80,000 BDT\n");
            break;
        default:
            printf("Your Salary: 50,000 BDT\n");
    }

    return 0;
}
```

```
#include <stdio.h>
int main() {
    int rank;
    printf("Enter your rank: ");
    scanf("%d", &rank);
    switch (rank) {
        case 1:
            printf("Your Salary: 2,50,000 BDT\n");
            break;
        case 2:
            printf("Your Salary: 2,10,000 BDT\n");
    }
}
```

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter your rank: 6
Your Salary: 50,000 BDT
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

35. Take two integers indicating the x and y coordinate of a two-dimensional graph paper where the center point is x = 0 and y = 0. Now print the quadrant of the given point. [If user gives input (4,5) you should print 'First quadrant'; If user gives input (-4, -5) you should print 'Third quadrant']

```
#include <stdio.h>
```

```
int main() {
    int x, y;
    printf("Enter the coordinates (x, y): ");
    scanf("%d %d", &x, &y);
```

```
if (x > 0 && y > 0) {
    printf("First quadrant\n");
} else if (x < 0 && y > 0) {
    printf("Second quadrant\n");
} else if (x < 0 && y < 0) {
```

```

printf("Third quadrant\n");

} else if (x > 0 && y < 0) {

    printf("Fourth quadrant\n");

} else {

    printf("On the axis\n");
}

}

return 0;
}

```

The screenshot shows a code editor interface with a dark theme. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. Below the menu, there are tabs for various files: Calculation.java, cal1.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (which is the active tab), Car.java, and Student.java. The main code area contains the following C code:

```

4 int x, y;
5
6 printf("Enter the coordinates (x, y): ");
7 scanf("%d %d", &x, &y);
8
9 if (x > 0 && y > 0) {
10     printf("First quadrant\n");
11 } else if (x < 0 && y > 0) {
12     printf("Second quadrant\n");
13 } else if (x < 0 && y < 0) {
14     printf("Third quadrant\n");
15 } else if (x > 0 && y < 0) {
16     printf("Fourth quadrant\n");
17 }

```

Below the code editor is a terminal window showing the command-line interface. The prompt PS C:\Users\fahim\OneDrive\Documents\A java practise> indicates the current working directory. The user runs the command cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if (\$?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (\$?) { .\tempCodeRunnerFile }. The terminal then prompts for coordinates: Enter the coordinates (x, y): 5 6. The output shows 'First quadrant'.

36. Print the EVEN numbers between 1 to 100 using loop

```
#include <stdio.h>

int main() {

for (int i = 2; i <= 100; i += 2) {

    printf("%d ", i);
}

```

```
}
```

```
printf("\n");
```

```
return 0;
```

```
}
```

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar. The title bar says "A java practise". The left sidebar has icons for files like Calculation.java, call.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (which is the active tab), Car.java, and Student.java. The main editor area contains the following C code:

```
#include <stdio.h>
int main() {
    for (int i = 2; i <= 100; i += 2) {
        printf("%d ", i);
    }
    printf("\n");
    return 0;
}
```

Below the editor, the terminal window shows the output of running the code:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50 52 54 56 58 60 62 64 66 68 70 72 74 76 78 80 82 84 86 88 90 92 94 96
98 100
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The bottom status bar shows file paths, line numbers, and other system information.

37. Print the numbers between 100 to 200 which are completely divisible by 3 and 5.

```
#include <stdio.h>
```

```
int main() {

    for (int i = 100; i <= 200; ++i) {

        if (i % 3 == 0 && i % 5 == 0) {

            printf("%d ", i);

        }

    }

    printf("\n");

}
```

```
return 0;
```

```
}
```

```
#include <stdio.h>
int main() {
    for (int i = 100; i <= 200; ++i) {
        if (i % 3 == 0 && i % 5 == 0) {
            printf("%d ", i);
        }
    }
    printf("\n");
}
return 0;
```

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
105 120 135 150 165 180 195
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

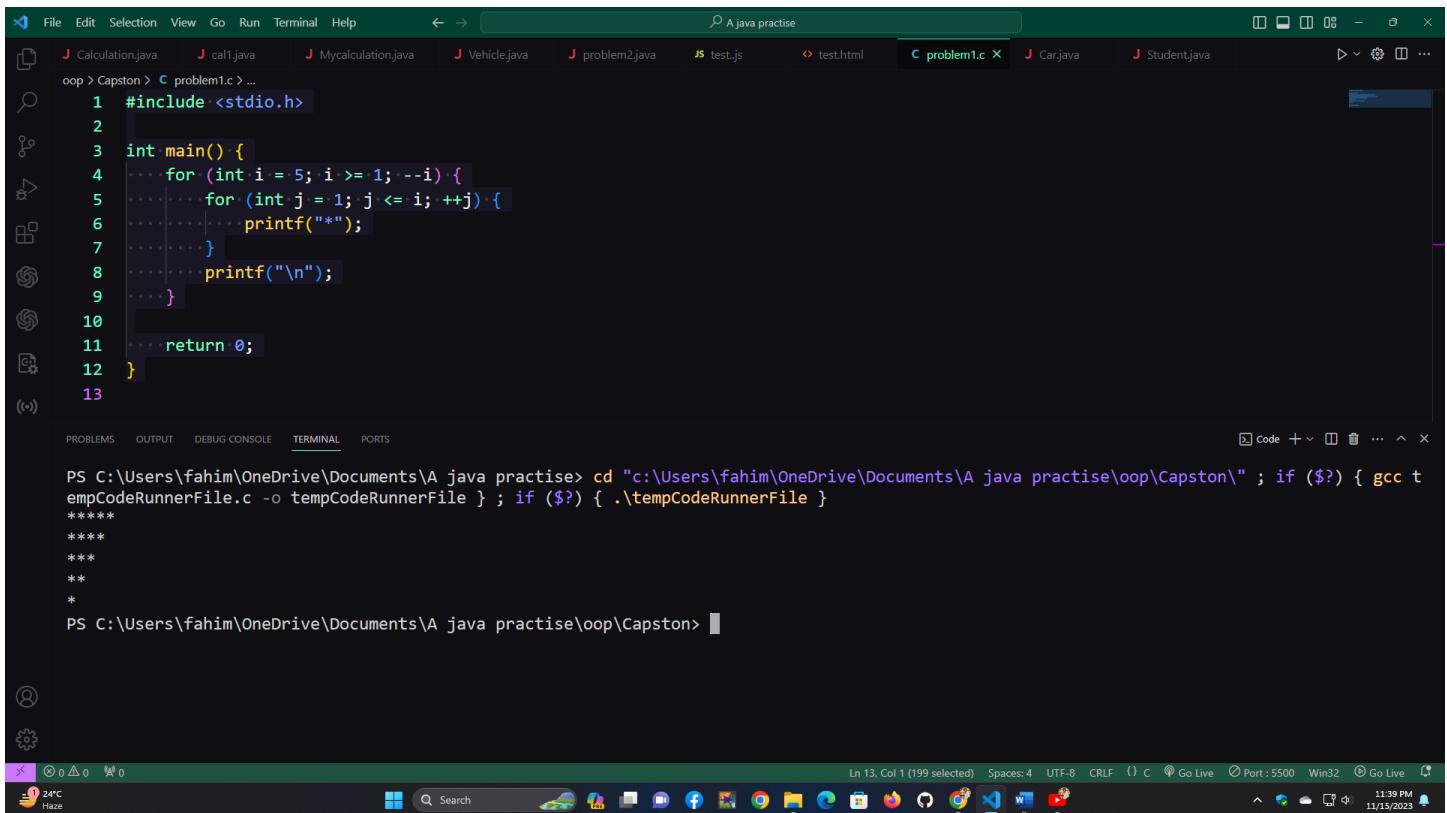
38. Write a program to print the following output using loop. ***** ***** *** ** *

```
#include <stdio.h>
```

```
int main() {
    for (int i = 5; i >= 1; --i) {
        for (int j = 1; j <= i; ++j) {
            printf("*");
        }
        printf("\n");
    }
}
```

```
return 0;
```

```
}
```



```
#include <stdio.h>
int main() {
    for (int i = 5; i >= 1; --i) {
        for (int j = 1; j <= i; ++j) {
            printf("*");
        }
        printf("\n");
    }
    return 0;
}
```

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
*****
***
```

39. Take an integer as input and print it's multiplication table up to 10. If user gives 5, your output should look like the following example
5 * 1 = 5 5 * 2 = 10 5 * 10 = 50

```
#include <stdio.h>
```

```
int main() {
```

```
    int number;
```

```
    printf("Enter an integer: ");
```

```
    scanf("%d", &number);
```

```
    for (int i = 1; i <= 10; ++i) {
```

```
        printf("%d * %d = %d\n", number, i, number * i);
```

```
}
```

```
return 0;
```

```
}
```

```
File Edit Selection View Go Run Terminal Help ↶ 🔍 A java practise
```

```
J CalculationJava J cal1.java J MycalculationJava J Vehicle.java J problem2.java JS test.js ⌂ test.html C problem1.c X J Carjava J Student.java
```

```
CalculationJava J cal1.java J MycalculationJava J Vehicle.java J problem2.java JS test.js ⌂ test.html C problem1.c X J Carjava J Student.java
```

```
oop > Capston > C problem1.c ...
```

```
3 int main() {
```

```
4     int number;
```

```
5
```

```
6     printf("Enter an integer: ");
```

```
7     scanf("%d", &number);
```

```
8
```

```
9     for (int i = 1; i <= 10; ++i) {
```

```
10        printf("%d * %d = %d\n", number, i, number * i);
```

```
11    }
```

```
12
```

```
13    return 0;
```

```
14 }
```

```
15
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
```

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
```

```
Enter an integer: 5
```

```
5 * 1 = 5
```

```
5 * 2 = 10
```

```
5 * 3 = 15
```

```
5 * 4 = 20
```

```
5 * 5 = 25
```

```
5 * 6 = 30
```

```
5 * 7 = 35
```

```
5 * 8 = 40
```

```
5 * 9 = 45
```

```
5 * 10 = 50
```

```
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

```
24°C Haze
```

```
Search
```

```
Ln 15, Col 1 (241 selected) Spaces:4 UTF-8 CRLF ⌂ C ⌂ Go Live ⌂ Port:5500 Win32 ⌂ Go Live ⌂
```

```
11:40 PM 11/15/2023
```

40. Write a program which will print the summation of the given series $101 + 99 + 97 + \dots + 3 + 1 = ?$

```
#include <stdio.h>
```

```
int main() {
```

```
    int sum = 0;
```

```
    for (int i = 101; i >= 1; i -= 2) {
```

```
        sum += i;
```

```
}
```

```
    printf("Summation: %d\n", sum);
```

```
    return 0;
```

}

A screenshot of the Visual Studio Code interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar. The left sidebar has icons for file types like Java (.java), C/C++ (.c), JavaScript (.js), and HTML (.html). The main editor window shows a C program named 'problem1.c' with the following code:

```
#include <stdio.h>
int main() {
    int sum = 0;
    for (int i = 101; i >= 1; i -= 2) {
        sum += i;
    }
    printf("Summation: %d\n", sum);
    return 0;
}
```

The terminal tab at the bottom shows the command-line output of running the program:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Summation: 2601
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

41. Write a program to print the following output using loop- * Blank Line *** Blank Line **** *
Blank Line *****

```
#include <stdio.h>
```

```
int main() {
```

```
for (int i = 1; i <= 5; ++i) {
```

```
for (int j = 1; j <= i * 2 - 1; ++j) {
```

```
printf("*");
```

• 60

```
return 0;
```

}

A screenshot of the Visual Studio Code interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar. The left sidebar contains icons for file types like Java, C, JavaScript, and HTML. The main editor area shows a C program named 'problem1.c' with the following code:

```
#include <stdio.h>
int main() {
    for (int i = 1; i <= 5; ++i) {
        for (int j = 1; j <= i * 2 - 1; ++j) {
            printf("*");
        }
        printf("\n");
        printf("Blank Line\n");
    }
    return 0;
}
```

The terminal below shows the output of running the program:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
*
Blank Line
***
Blank Line
*****
Blank Line
*****
Blank Line
*****
Blank Line
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

42. Write a program to print the following output using loop

```
#include <stdio.h>
```

```
int main() {
```

```
for (int i = 5; i >= 1; --i) {
```

```
for (int j = 1; j <= i; ++j) {
```

```
printf("%d ", j);
```

}

```
printf("\n");
```

}

```
return 0;
```

}

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar has icons for various files: Calculation.java, call.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (which is the active tab), Car.java, and Student.java. Below the sidebar are tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (which is selected), and PORTS. The main code editor window contains the following C code:

```
#include <stdio.h>
int main() {
    for (int i = 5; i >= 1; --i) {
        for (int j = 1; j <= i; ++j) {
            printf("%d ", j);
        }
        printf("\n");
    }
    return 0;
}
```

The terminal below the editor shows the output of running the program:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The taskbar at the bottom of the screen includes icons for weather (24°C Haze), search, file explorer, task manager, and browser, along with system status indicators like battery level, signal strength, and date/time (11/15/2023, 11:42 PM).

43. Write a program to display "A" to "Z" using loop.

```
#include <stdio.h>

int main() {

    for (char ch = 'A'; ch <= 'Z'; ++ch) {

        printf("%c ", ch);

    }

    printf("\n");

}

return 0;
```

```
#include <stdio.h>
int main() {
    for (char ch = 'A'; ch <= 'Z'; ++ch) {
        printf("%c ", ch);
    }
    printf("\n");
    return 0;
}
```

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

44. Print the summation of squares of all numbers from 5 to 25. $25 + 36 + 49 + 64 \dots + 576 + 625 = ?$

```
#include <stdio.h>
```

```
int main() {
```

```
    int sum = 0;
```

```
    for (int i = 5; i <= 25; ++i) {
```

```
        sum += i * i;
```

```
}
```

```
    printf("Summation: %d\n", sum);
```

```
    return 0;
```

```
}
```

A screenshot of the Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar shows several open files: Calculation.java, call.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (which is the active tab), Car.java, and Student.java. The main editor area contains the following C code:

```
#include <stdio.h>
int main() {
    int sum = 0;
    for (int i = 5; i <= 25; ++i) {
        sum += i * i;
    }
    printf("Summation: %d\n", sum);
    return 0;
}
```

The terminal below the editor shows the command-line output of running the program:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Summation: 5495
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The bottom status bar displays various system and application icons.

45. Write a program which will take an integer N from user and print all the numbers between N to –

```
#include <stdio.h>
```

```
int main() {
```

```
    int N;
```

```
    printf("Enter an integer N: ");
```

```
    scanf("%d", &N);
```

```
    while (N != -1) {
```

```
        printf("%d\n", N);
```

```
        printf("Enter an integer N: ");
```

```
        scanf("%d", &N);
```

```
}
```

```
return 0;
```

```
}
```

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar has icons for files like Calculation.java, cal1.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (which is the active tab), Car.java, and Student.java. The main code editor window contains the following C code:

```
oop > Capston > C problem1.c > main()
1 #include <stdio.h>
2
3 int main() {
4     int N;
5
6     printf("Enter an integer N: ");
7     scanf("%d", &N);
8
9     while (N != -1) {
10        printf("%d\n", N);
11        printf("Enter an integer N: ");
12        scanf("%d", &N);
13    }
14}
```

Below the code editor, there are tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (which is selected), and PORTS. The terminal window displays the following output:

```
Enter an integer N: 2
2
Enter an integer N: 3
3
Enter an integer N: 1
1
Enter an integer N: 4
4
Enter an integer N: 1
1
Enter an integer N: 1
1
Enter an integer N: -1
```

The status bar at the bottom shows the path 'PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>', file statistics (Ln 5, Col 1, Spaces: 4, CRLF), and system information (Port: 5500, Win32, Go Live, 11:45 PM, 11/15/2023).

32. Stop your program when user will give N= -1 as input.

```
#include <stdio.h>
```

```
int main() {
```

```
    int N, first = 0, second = 1, next;
```

```
    printf("Enter an integer N: ");
```

```
    scanf("%d", &N);
```

```
    printf("Fibonacci series up to %d: ", N);
```

```
    printf("%d %d ", first, second);
```

```
    for (int i = 2; i < N; ++i) {
```

```

next = first + second;

printf("%d ", next);

first = second;

second = next;

}

```

```
printf("\n");
```

```
return 0;
```

```
}
```

The screenshot shows a code editor interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Title Bar:** A java practise
- Toolbar:** Includes icons for file operations like Open, Save, Print, and a search bar.
- Left Sidebar:** Shows a tree view of files and folders, including Calculation.java, cal1.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (the active file), Car.java, and Student.java.
- Code Editor:**

```

1 #include <stdio.h>
2
3 int main() {
4     int N, first = 0, second = 1, next;
5
6     printf("Enter an integer N: ");
7     scanf("%d", &N);
8
9     printf("Fibonacci series up to %d: ", N);
10    printf("%d %d ", first, second);
11
12    for (int i = 2; i < N; ++i) {
13        next = first + second;
14        printf("%d ", next);
15    }
16
17    return 0;
18}
```
- Terminal:**

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter an integer N: 2
Fibonacci series up to 2: 0 1
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```
- Bottom Status Bar:** Shows file path (C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\problem1.c), line (Ln 11, Col 1), column (Spaces: 4), encoding (UTF-8), end-of-line (CRLF), character set (C), Go Live, Port 5500, Win32, Go Live, and system status (24°C Haze).

46. Write a program which will take an integer N as input from user and print Fibonacci series up to N. If user gives input N = 15, you should print 0 1 1 2 3 5 8 13

```
#include <stdio.h>
```

```
int main() {
```

```
int N, first = 0, second = 1, next;
```

```
printf("Enter an integer N: ");
```

```
scanf("%d", &N);
```

```
printf("Fibonacci series up to %d: ", N);
```

```
printf("%d %d ", first, second);
```

```
for (int i = 2; i < N; ++i) {
```

```
    next = first + second;
```

```
    printf("%d ", next);
```

```
    first = second;
```

```
    second = next;
```

```
}
```

```
printf("\n");
```

```
return 0;
```

```
}
```

A screenshot of the Visual Studio Code interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar shows several open files: Calculation.java, call.java, Mycalculation.java, Vehicle.java, problem2.java, test.js, test.html, problem1.c (the active file), Car.java, and Student.java. The main editor area contains the following C code:

```
#include <stdio.h>
int main() {
    int N, first = 0, second = 1, next;
    printf("Enter an integer N: ");
    scanf("%d", &N);
    printf("Fibonacci series up to %d: ", N);
    printf("%d %d ", first, second);
    for (int i = 2; i < N; ++i) {
        next = first + second;
        printf("%d ", next);
        first = second;
        second = next;
    }
}
```

The terminal below shows the command line and the output of the program:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter an integer N: 9
Fibonacci series up to 9: 0 1 1 2 3 5 8 13 21
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The bottom status bar shows file paths, line numbers, and other system information.

47. Write a program to print the following output in the given format using loop
1 1 12 21 123 321 1234 4321
12345 54321

```
#include <stdio.h>
```

```
int main() {
    for (int i = 1; i <= 5; ++i) {
        for (int j = 1; j <= i; ++j) {
            printf("%d", j);
        }
        printf("\n");
    }
}
```

```
return 0;
```

```
}
```

A screenshot of a Windows desktop environment. At the top is the taskbar with several pinned icons. Below it is a terminal window in Visual Studio Code. The terminal shows a command-line session:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
1
12
123
1234
12345
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

The terminal window has tabs at the bottom labeled PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS. The TERMINAL tab is selected. The status bar at the bottom of the terminal window displays file paths, line numbers, and other system information.

48. Find summation and average of all the numbers which are completely divisible by 3, 5 and 12 between 10 – 500. [Like 60 is divisible by all of those]

```
#include <stdio.h>
```

```
int main() {
    int sum = 0, count = 0;
```

```
    for (int i = 10; i <= 500; ++i) {
        if (i % 3 == 0 && i % 5 == 0 && i % 12 == 0) {
```

```
            sum += i;
```

```
            count++;
```

```
}
```

```
}
```

```
printf("Summation: %d\n", sum);
```

```
if (count > 0) {  
    printf("Average: %.2f\n", (float)sum / count);  
}  
else {  
    printf("No numbers found.\n");  
}  
  
return 0;  
}
```

A screenshot of the Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar containing "A java practise". The left sidebar features icons for file operations like Open, Save, Find, and Settings. The main workspace shows a C code editor with the following content:

```
1 #include <stdio.h>
2
3 int main() {
4     int sum = 0, count = 0;
5
6     for (int i = 10; i <= 500; ++i) {
7         if (i % 3 == 0 && i % 5 == 0 && i % 12 == 0) {
8             sum += i;
9             count++;
10        }
11    }
12
13    printf("Summation: %d\n", sum);
14    if (count > 0) {

```

The status bar at the bottom displays the terminal output:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Summation: 2160
Average: 270.00
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```

At the bottom right, there are system status icons for battery, signal, and time (11:48 PM, 11/15/2023).

49. Write a program to produce the following output using loop
1 2 4 3 6 9 4 8 12 16 5 10 15 20 25 6 12 18 24
30 36

```
#include <stdio.h>
```

```
int main() {
```

```
for (int i = 1; i <= 6; ++i) {  
    for (int j = 1; j <= i; ++j))
```

```

    printf("%d ", i * j);

}

printf("\n");

}

return 0;
}

```

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows multiple files including 'Calculation.java', 'Capston.java', 'MyCalculation.java', 'Vehicle.java', 'problem2.java', 'test.js', 'test.html', 'problem1.c' (the active file), 'Car.java', and 'Student.java'.
- Terminal:** Displays the command-line output of the program. It shows the code being compiled with 'gcc' and then executed. The output is a triangular multiplication table from 1 to 6.
- Bottom Status Bar:** Shows system information like battery level (0%), exchange rate (GBP/USD -0.65%), and system time (11:48 PM 11/15/2023).

```

1 #include <stdio.h>
2
3 int main() {
4     for (int i = 1; i <= 6; ++i) {
5         for (int j = 1; j <= i; ++j) {
6             printf("%d ", i * j);
7         }
8         printf("\n");
9     }
10    return 0;
11 }
12
13
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
1
2 4
3 6 9
4 8 12 16
5 10 15 20 25
6 12 18 24 30 36
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>

```

50. Write a program which will display all the prime numbers between 0 to N (N will be given by the user). If user gives N = 20, your code will print the following output 2 3 5 7 11 13 17 19

```

#include <stdio.h>

int isPrime(int num) {

if (num < 2) {

return 0; // not prime

}

```

```
for (int i = 2; i * i <= num; ++i) {  
    if (num % i == 0) {  
        return 0; // not prime  
    }  
}
```

```
return 1; // prime  
}
```

```
int main() {
```

```
    int N;
```

```
    printf("Enter an integer N: ");  
    scanf("%d", &N);
```

```
    printf("Prime numbers between 0 and %d: ", N);
```

```
    for (int i = 2; i <= N; ++i) {
```

```
        if (isPrime(i)) {
```

```
            printf("%d ", i);
```

```
        }
```

```
}
```

```
    printf("\n");
```

```
    return 0;
```

```
}
```

A screenshot of the Visual Studio Code (VS Code) interface. The top navigation bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'A java practise'. The left sidebar contains icons for file operations like Open, Save, Find, and Settings. The main editor area shows a Java file named 'problem1.c' with the following code:

```
5     return 0; // not prime
6 }
7
8 for (int i = 2; i * i <= num; ++i) {
9     if (num % i == 0) {
10         return 0; // not prime
11     }
12 }
13
14 return 1; // prime
15 }
16
17 int main() {
```

The bottom navigation bar has tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS, with TERMINAL being the active tab. The terminal window displays the following command-line session:

```
PS C:\Users\fahim\OneDrive\Documents\A java practise> cd "c:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston\" ; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerFile } ; if (?) { .\tempCodeRunnerFile }
Enter an integer N: 6
Prime numbers between 0 and 6: 2 3 5
PS C:\Users\fahim\OneDrive\Documents\A java practise\oop\Capston>
```