

Data Structures and algorithms

Submitted by:

Aryaveer Singh

102304064

3D12

Assignment 1

Q 1: Write a program to convert miles into kilometers (Km). Hint: 1 Mile = 1.609 Km. [Use macros, relevant name and types for variables].

```
#include <stdio.h>

#define MILE_TO_KM_CONVERSION_FACTOR 1.609

int main() {

    float miles;

    float kilometers;

    printf("Enter the distance in miles: ");

    scanf("%f", &miles);

    kilometers = miles * MILE_TO_KM_CONVERSION_FACTOR;

    printf("%.2f miles is equal to %.2f kilometers.\n", miles, kilometers);

    return 0;

}
```

Output

Time: 0.004ms

Memory: 2.83 Mb



```
Enter the distance in miles: 0.00 miles is equal to 0.00 kilometers.
```

Q 2: Write a program to find the number of positive, negative and zeros in a sequence of inputs (numbers) entered as data.

```
#include <stdio.h>

int main() {

    int num, positive_count = 0, negative_count = 0, zero_count = 0;

    char choice;

    do {

        printf("Enter a number: ");

        scanf("%d", &num);

        if (num > 0) {

            positive_count++;

        } else if (num < 0) {

            negative_count++;

        } else {

            zero_count++;

        }

        printf("Do you want to enter another number? (y/n): ");

        scanf(" %c", &choice);

    } while (choice == 'y' || choice == 'Y');


    printf("Number of positive numbers: %d\n", positive_count);

    printf("Number of negative numbers: %d\n", negative_count);

    printf("Number of zeros: %d\n", zero_count);

    return 0;

}
```

```
Output          Time: 0.003ms      Memory: 2.91 Mb    
Enter a number: Do you want to enter another number? (y/n): Number of  
positive numbers: 0  
Number of negative numbers: 0  
Number of zeros: 1
```

Q 3: Compute the tax due based on a tax table given below:

Program Input: Salary amount

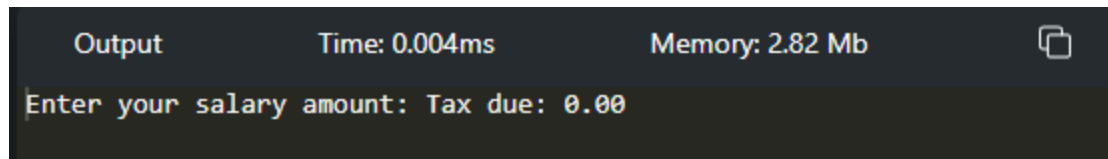
Program Output: Returns the tax due for $0.0 \leq \text{salary} \leq 150,000.00$; returns -1.0 if salary is outside the table range.

```
#include <stdio.h>  
  
int main() {  
    double salary, tax = -1.0;  
    printf("Enter your salary amount: ");  
    scanf("%lf", &salary);  
    if (salary < 0.0 || salary > 150000.00) {  
        printf("Tax due: %.2lf\n", tax);  
    } else if (salary <= 14999.99) {  
        tax = salary * 0.15;  
    } else if (salary <= 29999.99) {  
        tax = 2250.00 + (salary - 15000.00) * 0.18;  
    } else if (salary <= 49999.99) {  
        tax = 5400.00 + (salary - 30000.00) * 0.22;  
    } else if (salary <= 79999.99) {  
        tax = 11000.00 + (salary - 50000.00) * 0.27;  
    } else { salary <= 150000.00  
        tax = 21600.00 + (salary - 80000.00) * 0.33;  
    }  
}
```

```

if (tax != -1.0) {
    printf("Tax due: %.2lf\n", tax);
}
return 0;
}

```



The screenshot shows a terminal window with a dark background. At the top, there are three status bars: 'Output', 'Time: 0.004ms', and 'Memory: 2.82 Mb'. Below these, the program's output is displayed in a monospaced font: 'Enter your salary amount: Tax due: 0.00'. A small icon of a document with a plus sign is visible in the top right corner of the terminal window.

Q 4: Write an interactive program (menu driven) in 'C' (using functions) to compute the area of a selected geometrical figure (from a list of such figures (square, rectangle, and circle)).

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

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

Function to calculate area of a square

```

float areaOfSquare() {
    float side;

    printf("Enter the side length of the square: ");

    scanf("%f", &side);

    return side * side;
}

```

Function to calculate area of a rectangle

```

float areaOfRectangle() {
    float length, width;

    printf("Enter the length and width of the rectangle: ");

    scanf("%f %f", &length, &width);

    return length * width;
}

```

Function to calculate area of a circle

```
float areaOfCircle() {  
    float radius;  
    printf("Enter the radius of the circle: ");  
    scanf("%f", &radius);  
    return M_PI * radius * radius;  
}  
  
int main() {  
    int choice;  
    float area;  
    do {  
        printf("\nMenu:\n");  
        printf("1. Calculate area of a Square\n");  
        printf("2. Calculate area of a Rectangle\n");  
        printf("3. Calculate area of a Circle\n");  
        printf("4. Exit\n");  
        printf("Enter your choice: ");  
        scanf("%d", &choice);  
        switch (choice) {  
            case 1:  
                area = areaOfSquare();  
                printf("Area of the square: %.2f\n", area);  
                break;  
            case 2:  
                area = areaOfRectangle();  
                printf("Area of the rectangle: %.2f\n", area);  
                break;  
            case 3:  
                area = areaOfCircle();  
                printf("Area of the circle: %.2f\n", area);
```

```

        break;

    case 4:

        printf("Exiting program.\n");

        break;

    default:

        printf("Invalid choice. Please try again.\n");

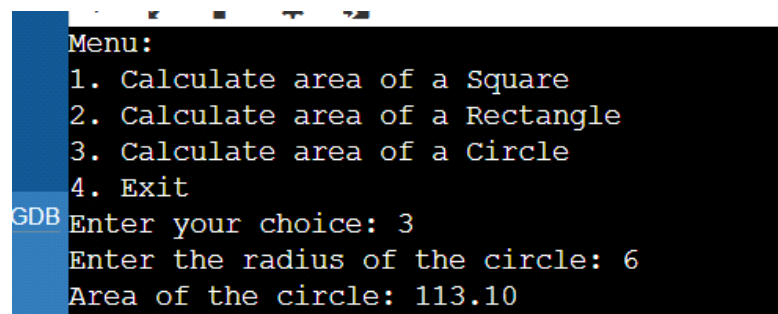
    }

} while (choice != 4);

return 0;

}

```



```

Menu:
1. Calculate area of a Square
2. Calculate area of a Rectangle
3. Calculate area of a Circle
4. Exit
GDB Enter your choice: 3
Enter the radius of the circle: 6
Area of the circle: 113.10

```

Q 5: Write a program to display first n elements of Fibonacci series.

```

#include <stdio.h>

int main() {

    int n, first = 0, second = 1, next, i;

    printf("Enter the number of Fibonacci elements to display: ");

    scanf("%d", &n);

    printf("Fibonacci series up to %d elements:\n", n);

    for (i = 0; i < n; i++) {

        if (i <= 1) {

            next = i;

        } else {

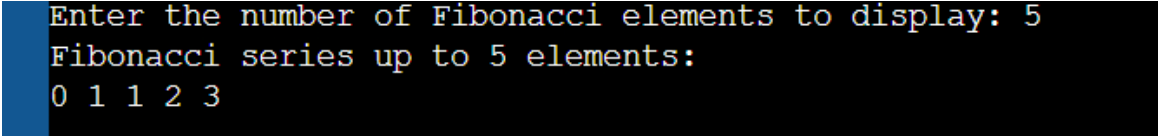
            next = first + second;


```

```

        first = second;
        second = next;
    }
    printf("%d ", next);
}
printf("\n");
return 0;
}

```



A terminal window with a black background and a blue vertical bar on the left. The text is displayed in a monospaced font with syntax highlighting: 'Enter the number of Fibonacci elements to display: 5' is in yellow, 'Fibonacci series up to 5 elements:' is in green, and the output '0 1 1 2 3' is in white.

```

Enter the number of Fibonacci elements to display: 5
Fibonacci series up to 5 elements:
0 1 1 2 3

```

Q 6: Write a program to print a table book from Table X to Table Y. X and Y are user inputs.

```

#include <stdio.h>

int main() {
    int start_table, end_table, i, j;
    printf("Enter the starting table number: ");
    scanf("%d", &start_table);
    printf("Enter the ending table number: ");
    scanf("%d", &end_table);
    for (i = start_table; i <= end_table; i++) {
        printf("\nMultiplication Table for %d:\n", i);
        for (j = 1; j <= 10; j++) {
            printf("%d x %d = %d\n", i, j, i * j);
        }
    }
    return 0;
}

```

```
}
```

```
Enter the starting table number: 4
```

```
Enter the ending table number: 5
```

```
Multiplication Table for 4:
```

```
4 x 1 = 4
```

```
4 x 2 = 8
```

```
4 x 3 = 12
```

```
4 x 4 = 16
```

```
4 x 5 = 20
```

```
4 x 6 = 24
```

```
4 x 7 = 28
```

```
4 x 8 = 32
```

```
4 x 9 = 36
```

```
4 x 10 = 40
```

```
Multiplication Table for 5:
```

```
5 x 1 = 5
```

```
5 x 2 = 10
```

```
5 x 3 = 15
```

```
5 x 4 = 20
```

```
5 x 5 = 25
```

```
5 x 6 = 30
```

```
5 x 7 = 35
```

```
5 x 8 = 40
```

```
5 x 9 = 45
```

```
5 x 10 = 50
```

Q 7: Write a program to compute factorial of a number using iterative approach.

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

```
int main() {
```

```
    int num, i;
```

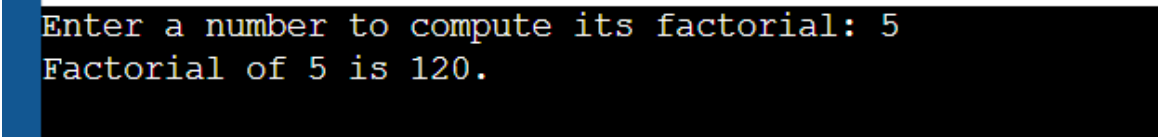
```
    long long factorial = 1;
```



```

printf("Enter a number to compute its factorial: ");
scanf("%d", &num);
if (num < 0) {
    printf("Factorial of a negative number doesn't exist.\n");
} else {
    for (i = 1; i <= num; i++) {
        factorial *= i;
    }
    printf("Factorial of %d is %lld.\n", num, factorial);
}
return 0;
}

```



```

Enter a number to compute its factorial: 5
Factorial of 5 is 120.

```

Q 8: Write a program to swap two numbers using functions.

```

#include <stdio.h>

void swap(int *a, int *b) {
    int temp = *a;
    *a = *b;
    *b = temp;
}

int main() {
    int num1, num2;

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

    printf("Before swapping: num1 = %d, num2 = %d\n", num1, num2);
}

```

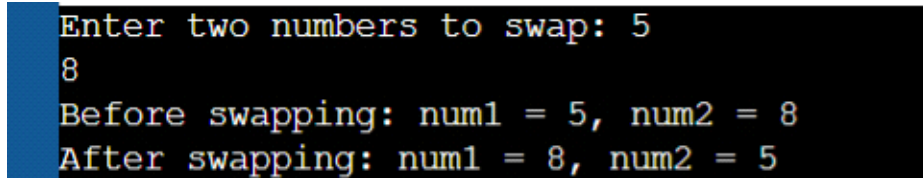
```

swap(&num1, &num2);

printf("After swapping: num1 = %d, num2 = %d\n", num1, num2);

return 0;
}

```



```

Enter two numbers to swap: 5
8
Before swapping: num1 = 5, num2 = 8
After swapping: num1 = 8, num2 = 5

```

Q 9: Write a function that returns the first integer between n_min and n_max entered as data to the calling function (main).

```

#include <stdio.h>

int findFirstInRange(int min, int max) {
    int num;
    while (1) {
        printf("Enter a number: ");
        scanf("%d", &num);
        if (num >= min && num <= max) {
            return num;
        }
        printf("Number is not in the range [%d, %d]. Please try again.\n", min, max);
    }
}

int main() {
    int n_min, n_max;
    printf("Enter the minimum and maximum range: ");
    scanf("%d %d", &n_min, &n_max);
    int first_in_range = findFirstInRange(n_min, n_max);
    printf("The first number entered within the range [%d, %d] is: %d\n", n_min, n_max, first_in_range);
}

```

```
    return 0;
}
```

```
Enter the minimum and maximum range: 5
100
Enter a number: 7
The first number entered within the range [5, 100] is: 7
```

Q 10: Write nests of loops that cause the following output to be displayed.

```
/*
0
0 1
0 1 2
0 1 2 3
0 1 2 3 4
0 1 2 3 4 5
0 1 2 3 4
0 1 2 3
0 1 2
0 1
0
*/
```

```
#include <stdio.h>

int main() {
    int i, j;

    Upper part of the pattern
    for (i = 0; i <= 5; i++) {
        for (j = 0; j <= i; j++) {
            printf("%d ", j);
        }
    }
}
```

```

        printf("\n");
    }
    Lower part of the pattern
    for (i = 4; i >= 0; i--) {
        for (j = 0; j <= i; j++) {
            printf("%d ", j);
        }
        printf("\n");
    }
    return 0;
}

```

```

0
0 1
0 1 2
0 1 2 3
0 1 2 3 4
0 1 2 3 4 5
0 1 2 3 4
0 1 2 3
0 1 2
0 1
0

```