LAB ASSIGNMENT 1
UEE410
Naysa Kukreja
102304049
3D13

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

```
#include<stdio.h>
int main(){
int mile;
printf("enter the number");
scanf("%d",&mile);
int km=mile*1.609;
printf("mile converted from km is:%d",km);
return 0;
}
```

Output

```
enter the number4
mile coverted from km is:6

=== Code Execution Successful ===
```

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 n, positive = 0, negative = 0, zero = 0;
printf("Number of inputs: ");
scanf("%d", &n);
int a[n];
printf("Enter %d inputs:\n", n);
for (int i = 0; i < n; i++) {
```

```
scanf("%d", &a[i]);
}

for (int i = 0; i < n; i++) {
if (a[i] > 0) {
positive++;
}
else if (a[i] < 0) {
negative++;
}
else {
zero++;
}
}
printf("Positive count: %d\n", positive);
printf("Negative count: %d\n", negative);
printf("Zero count: %d\n", zero);
return 0;
}
```

Output

```
Number of inputs: 2
Enter 2 inputs:
-2
-3
Positive count: 0
Negative count: 2
Zero count: 0


=== Code Execution Successful ===
```

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 <= salary <= 150,000.00; returns -1.0 if salary is outside the table range.

Salary Range ($)	Base Tax ($)	Percentage of Excess
0.00–14,999.99	0.00	15
15,000.00–29,999.99	2,250.00	18
30,000.00–49,999.99	5,400.00	22
50,000.00–79,999.99	11,000.00	27
80,000.00–150,000.00	21,600.00	33

```
#include <stdio.h>
double computeTax(double salary) {
if (salary < 0.0 || salary > 150000.00) {
return -1.0;
}
if (salary <= 14999.99) {
return 0.00 + (salary * 0.15);
}
else if (salary <= 29999.99) {
return 2250.00 + ((salary - 15000.00) * 0.18);
}
else if (salary <= 49999.99) {
return 5400.00 + ((salary - 30000.00) * 0.22);
}
else if (salary <= 79999.99) {
return 11000.00 + ((salary - 50000.00) * 0.27);
}
else {
return 21600.00 + ((salary - 80000.00) * 0.33);
}
}
int main() {
double salary, tax;
printf("Enter the salary amount: $");
scanf("%lf", &salary);
tax = computeTax(salary);
if (tax == -1.0) {
printf("Invalid salary: The amount is outside the table range.\n");
}
else {
printf("Tax due: $%.2f\n", tax);
```

```
}
return 0;
```

```
Output

Enter the salary amount: $
40000
Tax due: $7600.00


=== Code Execution Successful ===
```

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>
int triangle(int h,int b){
return(h*b*0.5);
}
int square(int s){
return(s*s);
}
int rectangle(int l,int b){
return(l*b);
}
int circle(int r){
return(3.14*r*r);
}
int main(){
int r,h,l,b,s;
int choice;
printf("enter the choice \n1.circle,\n2.sqaure,\n3.rectangle,\n4.triangle\n");
scanf("%d",&choice);
if(choice==1){
printf("enter the parameters(radius) of circle:");
```

```
scanf("%d",&r);
int carea=circle(r);
printf("the area of circle is: %d", carea );
}
else if(choice==2){
printf("enter the parameters(side) of square:");
scanf("%d",&s);
int sarea=square(s);
printf("the area of square is: %d", sarea );
}
else if(choice==3){
printf("enter the parameters(length,breadth) of rectangle(comma separated):\n");
scanf("%d,%d",&l,&b);
int rarea=rectangle(l,b);
printf("the area of rectangle is: %d", rarea );
}
else if(choice==4){
printf("enter the parameters(base,height) of triangle(comma separated):\n");
scanf("%d,%d",&b,&h);
int tarea=triangle(h,b);
printf("the area of triangle is: %d", tarea );
}
return 0;
}
```

Output Clear

```
enter the choice
1.circle,
2.sqaure,
3.rectangle,
4.triangle
4
enter the parameters(base,height) of triangle(comma  seperated):
2,5
the area of triangle is: 5

=== Code Execution Successful ===
```

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

```
#include <stdio.h>
int main() {
int n, i;
int t1 = 0, t2 = 1;
int nextTerm;

printf("Enter the number of terms: ");
scanf("%d", &n);
printf("Fibonacci Series: ");

for (i = 1; i <= n; ++i) {
printf("%d, ", t1);
nextTerm = t1 + t2;
t1 = t2;
t2 = nextTerm;
}
return 0;
}
```

```
Output

Enter the number of terms: 5
Fibonacci Series: 0, 1, 1, 2, 3,

=== Code Execution Successful ===
```

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 (X): ");
scanf("%d", &start_table);

printf("Enter the ending table number (Y): ");
scanf("%d", &end_table);

if (end_table < start_table) {
printf("Error: The ending number cannot be less than the starting number.\n");
```

```
return 1;
}

for (i = start_table; i <= end_table; ++i) {
printf("\nMultiplication Table of %d:\n", i);

for (j = 1; j <= 10; ++j) {
printf("%d * %d = %d\n", i, j, i * j);
}
}
return 0;
}
```

Output

```
Enter the starting table number (X): 6
Enter the ending table number (Y): 8

Multiplication Table of 6:
6 * 1 = 6
6 * 2 = 12
6 * 3 = 18
6 * 4 = 24
6 * 5 = 30
6 * 6 = 36
6 * 7 = 42
6 * 8 = 48
6 * 9 = 54
6 * 10 = 60

Multiplication Table of 7:
7 * 1 = 7
7 * 2 = 14
7 * 3 = 21
7 * 4 = 28
7 * 5 = 35
7 * 6 = 42
7 * 7 = 49
7 * 8 = 56
7 * 9 = 63
7 * 10 = 70
```

```
Multiplication Table of 8:
8 * 1 = 8
8 * 2 = 16
8 * 3 = 24
8 * 4 = 32
8 * 5 = 40
8 * 6 = 48
8 * 7 = 56
8 * 8 = 64
8 * 9 = 72
8 * 10 = 80


=== Code Execution Successful ===
```

7.Write a program to compute the factorial of a number using an iterative approach.

```
#include <stdio.h>

int main(){
int num;
int fact=1;
printf("enter the number for factorial\n");
scanf("%d",&num);
for(int i=1;i<num+1;i++){
fact=fact*i;
}
printf("%d",fact);
return 0;
}
```

Output

```
enter the number for factorial
6
720

=== Code Execution Successful ===
```

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 a;
int b;
printf("enter the number a and b (comma separated) \n");
scanf("%d,%d",&a,&b);
printf("before swap %d,%d\n",a,b);
swap(&a,&b);
printf("after swap\n%d,%d",a,b);
return 0;
}
```

Output

```
enter the number a and b (comma separated)
5,6
before swap 5,6
after swap
6,5

=== Code Execution Successful ===
```

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 findFirstIntegerInRange(int n_min, int n_max) {
int input_num;
printf("Please enter integers. The first one between %d and %d will be returned.\n", n_min,
n_max);

while (1) {
printf("Enter a number: ");
scanf("%d", &input_num);

if (input_num >= n_min && input_num <= n_max) {
return input_num; // Return the valid number
}
else {
printf("The number is not within the range. Try again.\n");
}
}
}

int main() {
int min_val, max_val, found_number;

printf("Enter the minimum value of the range: ");
scanf("%d", &min_val);
printf("Enter the maximum value of the range: ");
scanf("%d", &max_val);

found_number = findFirstIntegerInRange(min_val, max_val);
printf("\nSuccess! The first integer found in the range was: %d\n", found_number);
return 0;
}
```

Output Clear

```
Enter the minimum value of the range: 40
Enter the maximum value of the range: 70
Please enter integers. The first one between 40 and 70 will be returned.
Enter a number: 60

Success! The first integer found in the range was: 60


=== Code Execution Successful ===
```

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;
for (i = 0; i <= 5; i++) {
for (j = 0; j <= i; j++) {
printf("%d ", j);
}
printf("\n");
}
for (i = 4; i >= 0; i--) {
for (j = 0; j <= i; j++) {
printf("%d ", j);
}
printf("\n");
}
return 0;
}
```

Output

```
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


=== Code Execution Successful ===
```