

Table of Contents

CHAPTER	TOPIC	PAGE
1	1.1 Introduction	3
	1.2 Design Goals/Objective	4
2	IMPLEMENTATION OF THE PROJECT	5-25
	2.1 Flow Chart	5
	2.2 Implementation	6-22
	2.3 Outputs	23-25
3	CONCLUSION	
	Learning Outcome	26
	Future Scope	
4	Reference	26

Chapter 1

Introduction

1.1 Introduction

C is a procedural programming language. It was initially developed by Dennis Ritchie in the year 1972. It was mainly developed as a system programming language to write an operating system. The main features of the C language include low-level memory access, a simple set of keywords, and a clean style, these features make C language suitable for system programming's like an operating system or compiler development.

Many later languages have borrowed syntax/features directly or indirectly from the C language. Like syntax of Java, PHP, JavaScript, and many other languages are mainly based on the C language.

'C' is a powerful programming language which is strongly associated with the UNIX operating system. Even most of the UNIX operating system is coded in 'C'. Initially 'C' programming was limited to the UNIX operating system, but as it started spreading around the world, it became commercial, and many compilers were released for cross-platform systems. Today 'C' runs under a variety of operating systems and hardware platforms. As it started evolving many different versions of the language were released. At times it became difficult for the developers to keep up with the latest version as the systems were running under the older versions. To assure that 'C' language will remain standard, American National Standards Institute (ANSI) defined a commercial standard for 'C' language in 1989. Later, it was approved by the International Standards Organization (ISO) in 1990. 'C' programming language is also called as 'ANSI C'.

Why learn C Language?

As we studied earlier, 'C' is a base language for many programming languages. So, learning 'C' as the main language will play an important role while studying other programming languages. 'C' is a structured programming language in which program is divided into various modules. Each module can be written separately and together it forms a single 'C' program.

- It shares the same concepts such as data types, operators, control statements and many more.
- 'C' can be used widely in various applications. It is a simple language and provides faster execution.
- This structure makes it easy for testing, maintaining and debugging processes.
- 'C' contains 32 keywords, various data types and a set of powerful built-in functions that make programming very efficient.

1.2 Design Goals/Objective

The goal of the project is to build a Calculator.

Here we will build a program for calculating things. It will not be a regular calculator as we use in our daily life but it will be a futuristic calculator where you can calculate anything & everything withing a single calculator from anywhere and anytime.

User can choose whatever kind of calculation he/she wants to do. By their choice there will be other options related to their chosen options.

The program is basically for everyone, every kind of people will use it. So, we have made the design as simple as it can be.

The main object for this project is to help different people from different field with their calculation within their daily life activities. It's a very common problem that there are several times when we need to do some calculation and at that moment we will suffer through a problem with no pen-paper or the exact formula will not be in our mind. So, what do we do at this moment? We just try to search the solution from online or trying to get pen-paper and then try to calculate the problem manually. In this way we will waste a lot of time and also sometimes there will be some issue that the online formula or system will not be correct. So, yes, at this time our project comes with a simple and workable solution. Our project will let you know the perfect answer/result with the perfect formula.

Chapter 2

Implementation of the Project

2.1 Flow Chart

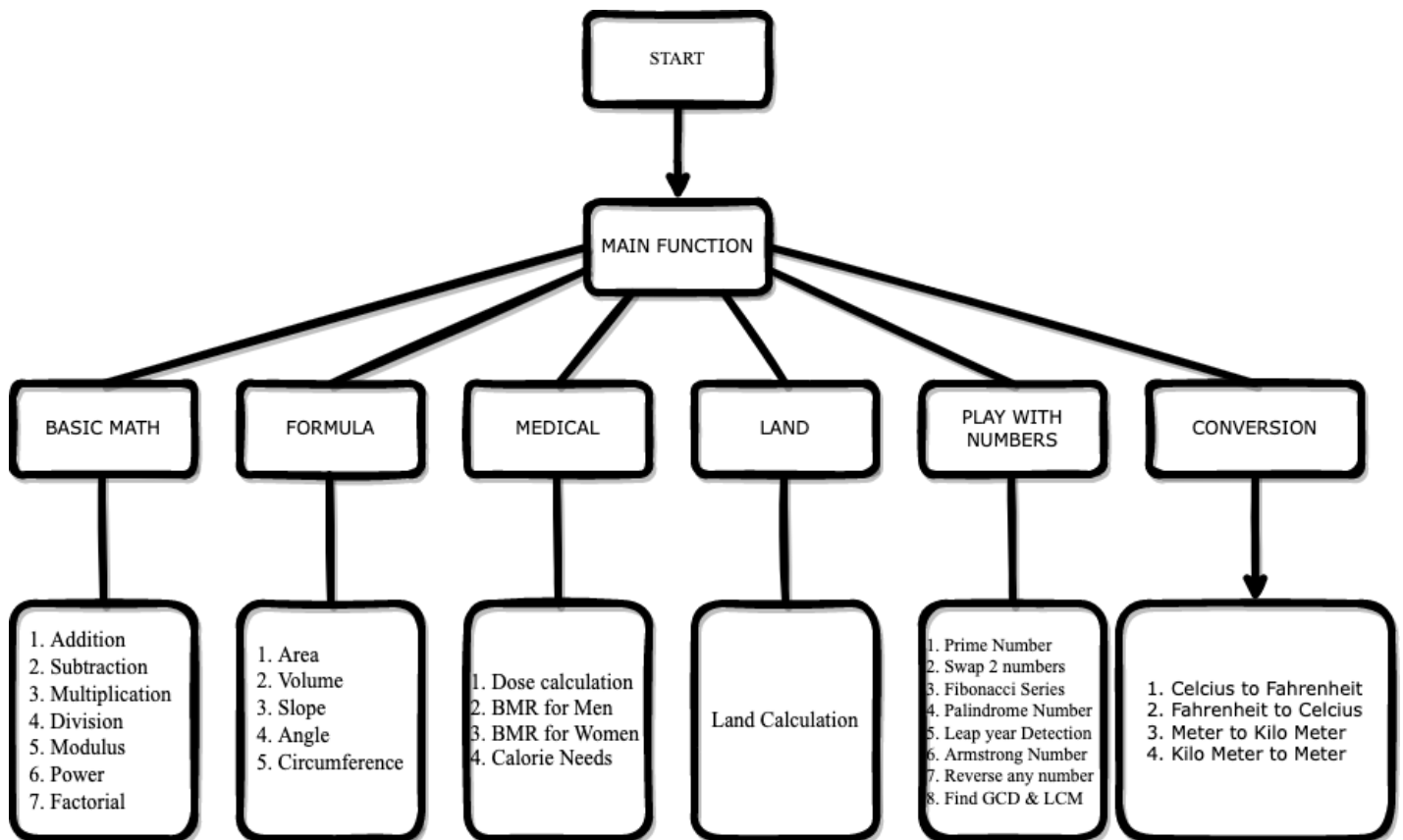


Figure1: Flow Chart of the whole program in short term

2.2 Implementation

C source code

```
1. #include<stdio.h>
2. #include<stdio.h>
3. #include<math.h>
4. #include<string.h>
5. #include<time.h>
6. #include <stdlib.h>
7.
8. struct choice{
9.     char choice_char;
10.    int choice_int;
11.    float choice_float;
12. };
13. struct choice ch;
14.
15. void calculator_op();
16. void basic_math();
17. void formula();
18. void area();
19. void volume();
20. void medical();
21. void play_math();
22. void conversion();
23.
24. void addition();
25. void subtraction();
26. void multiplication();
27. void division();
28. void modulus();
29. void power();
30. int factorial();
31.
32. void square();
33. void rectangle();
34. void triangle();
35. void circle();
36. void trapezoid();
37. void ellipse();
38. void cylinder_volume();
39. void cube_volume();
40. void cuboid_volume();
41. // void square_volume();
```

```

42. // void sphere_volume();
43. void slope();
44.
45. void mbr_women();
46. void mbr_men();
47. void dose_cal();
48. void cal_need();
49. void cal_men();
50. void cal_women();
51.
52. void prime_num();
53. void swap();
54. void fibonacci();
55. void palindrome();
56. void leap_year();
57. void armstrong_num();
58. void reverse_num();
59. void gcd_lcm();
60.
61. void c_f();
62.
63. int main(){
64.     calculator_op();
65.     scanf("%d", &ch.choice_int);
66.     switch (ch.choice_int) {
67.         case 1:basic_math();
68.             break;
69.         case 2:formula();
70.             break;
71.         case 3:medical();
72.             break;
73.         case 4:calculator_op();
74.             break;
75.         case 5:play_math();
76.             break;
77.         case 6:conversion();
78.             break;
79.         default:
80.             printf("***** Invalid Input *****\nRestart the Program\nPlease try to give
            values from 1 to 5");
81.             calculator_op();
82.             break;
83.     }
84.     return 0;
85. }
86.
87.

```

```

88.          // // // FUNCTION DEFINATIONS // // //
89.
90. void calculator_op(){
91.     printf("\n\n\n\n");
92.     printf("          ----- "CALCULATOR FOR EVERYONE" ----- \n");
93.     printf("          ----- ONE CALCULATOR CAN DO 1000 TYPES OF CALCULATION ----- \n\n");
94.     printf("          *** WELCOME TO OUR MODERN CALCULATOR ***");
95.
96.     struct tm* ptr;
97.     time_t t;
98.     t = time(NULL);
99.     ptr = localtime(&t);
100.    printf("\n          %s", asctime(ptr));
101.
102.    printf("\n          ***** \n
          *****");
103.    printf("\n\nChoose any of these type of Calculation you want to do: \n\n1. Basic Math(+, -, *, /,
^)\n2. Area, Volume, Slope and so more\n3. Madical Field\n4. Land Calculation\n5. Play with number by
calculation\n6. Conversion\n\nInput: ");
104. }
105. void basic_math(){
106.    printf("*** CHOSE ONE FROM THE LIST\n\n");
107.    printf("1. Addition \n");
108.    printf("2. Subtraction \n");
109.    printf("3. Multiplication \n");
110.    printf("4. Division \n");
111.    printf("5. Modulus\n");
112.    printf("6. Power \n");
113.    printf("7. Factorial \n\nInput: ");
114.    scanf("%d", &ch.choice_int);
115.    switch (ch.choice_int) {
116.    case 1:addition();
117.        break;
118.    case 2:subtraction();
119.        break;
120.    case 3:multiplication();
121.        break;
122.    case 4:division();
123.        break;
124.    case 5:modulus();
125.        break;
126.    case 6:power();
127.        break;
128.    case 7:factorial();
129.        break;
130.    default:

```



```

131.     printf("***** Invalid Input *****\nRestart the Program\nPlease try to give
        values from 1 to 7");
132.     basic_math();
133.     break;
134. }
135. }
136. void formula(){
137.     printf("\n\n*** CHOSE ONE FROM THE LIST ***\n\n");
138.     printf("1. Area \n");
139.     printf("2. Volume \n");
140.     printf("3. Slope \n");
141.     printf("4. Angle \n");
142.     printf("5. Circumference\n\nInput:");
143.     scanf("%d",&ch.choice_int);
144.     switch (ch.choice_int){
145.     case 1:area();
146.         break;
147.     case 2:volume();
148.         break;
149.     case 3: slope();
150.         break;
151.     default:
152.         printf("Please select from option 1 to 4");
153.         formula();
154.         break;
155.     }
156. }
157. void area(){
158.     printf("\n\n1. Square\n2. Rectangle\n3. Triangle\n4. Circle\n5. Trapezoid\n6. Ellipse\n\n");
159.     printf("Choose one from above options: ");
160.     scanf("%d", &ch.choice_int);
161.     switch (ch.choice_int){
162.     case 1:square();
163.         break;
164.     case 2:rectangle();
165.         break;
166.     case 3:triangle();
167.         break;
168.     case 4:circle();
169.         break;
170.     case 5:trapezoid();
171.         break;
172.     case 6:ellipse();
173.         break;
174.     default:
175.         printf("Please select from option 1 to 6");
176.         area();

```

```

177.         break;
178.     }
179. }
180. void volume(){
181.     printf("\n\n1. Cylinder\n2. Cube\n3. Rectangular Solid or Cuboid\n4. Square or Rectangular
    Pyramid\n5. Sphere\n\n");
182.     printf("Choose one from above options: ");
183.     scanf("%d", &ch.choice_int);
184.     switch (ch.choice_int){
185.         case 1:cylinder_volume();
186.             break;
187.         case 2:cube_volume();
188.             break;
189.         case 3:cuboid_volume();
190.             break;
191.         // case 4:square_volume();
192.         //     break;
193.         // case 5:sphere_volume();
194.         //     break;
195.         default:
196.             printf("Please select from option 1 to 5");
197.             volume();
198.             break;
199.     }
200. }
201. void slope(){
202.     printf("Enter Cordinates below: \n");
203.     printf("x1 = ");
204.     scanf("%f",&ch.choice_float);
205.     int x1 = ch.choice_float;
206.     printf("y1 = ");
207.     scanf("%f",&ch.choice_float);
208.     int y1 = ch.choice_float;
209.     printf("x2 = ");
210.     scanf("%f",&ch.choice_float);
211.     int x2 = ch.choice_float;
212.     printf("y2 = ");
213.     scanf("%f",&ch.choice_float);
214.     int y2 = ch.choice_float;
215.     float slope = (y2 - y1)/(x2 - x1);
216.     printf("The Slope m = %.4f", slope);
217. }
218. void medical(){
219.     printf("\n\n*** CHOSE ONE FROM THE LIST ***\n\n");
220.     printf("1. Dose calculation per weight prescribed per minute \n");
221.     printf("2. BMR for Men \n");
222.     printf("3. BMR for Women \n");

```

```

223.     printf("4. Calorie Needs \n\nInput: ");
224.     scanf("%d",&ch.choice_int);
225.     switch (ch.choice_int){
226.     case 1:dose_cal();
227.         break;
228.     case 2:mbr_men();
229.         break;
230.     case 3:mbr_women();
231.         break;
232.     case 4:cal_need();
233.         break;
234.     default:
235.         printf("Please select from option 1 to 3");
236.         medical();
237.         break;
238.     }
239. }

240. void conversion(){
241.     printf("\nChoose one from given options: \n");
242.     printf("1. Celcius to Fahrenheit\n");
243.     printf("2. Fahrenheit to Celcius\n");
244.     printf("3. Meter to Kilo Meter\n");
245.     printf("4. Kilo Meter to Meter\n");
246.     scanf("%d", &ch.choice_int);
247.     switch (ch.choice_int){
248.     case 1: c_f();
249.         break;
250.     default:
251.         conversion();
252.         break;
253.     }
254. }
255.

256. //////////// BASIC MATH CALCULATION ////////////
257. void addition()
258. {
259.     int n, total=0, k=0, number;
260.     printf("\nEnter the number of elements you want to add:");
261.     scanf("%d",&n);
262.     printf("Please enter %d numbers one by one: \n",n);
263.     while(k<n)
264.     {
265.         scanf("%d",&number);
266.         total=total+number;
267.         k=k+1;
268.     }

```

```

269.     printf("Sum of %d numbers = %d \n",n,total);
270. }
271. void subtraction()
272. {
273.     int a, b, c = 0;
274.     printf("\nPlease enter first number  : ");
275.     scanf("%d", &a);
276.     printf("Please enter second number : ");
277.     scanf("%d", &b);
278.     c = a - b;
279.     printf("\n%d - %d = %d\n", a, b, c);
280. }
281. void multiplication()
282. {
283.     int a, b, mul=0;
284.     printf("\nPlease enter first numb  : ");
285.     scanf("%d", &a);
286.     printf("Please enter second number: ");
287.     scanf("%d", &b);
288.     mul=a*b;
289.     printf("\nMultiplication of entered numbers = %d\n",mul);
290. }
291. void division()
292. {
293.     int a, b, d=0;
294.     printf("\nPlease enter first number  : ");
295.     scanf("%d", &a);
296.     printf("Please enter second number : ");
297.     scanf("%d", &b);
298.     d=a/b;
299.     printf("\nDivision of entered numbers=%d\n",d);
300. }
301. void modulus()
302. {
303.     int a, b, d=0;
304.     printf("\nPlease enter first number  : ");
305.     scanf("%d", &a);
306.     printf("Please enter second number : ");
307.     scanf("%d", &b);
308.     d=a%b;
309.     printf("\nModulus of entered numbers = %d\n",d);
310. }
311. void power()
312. {
313.     double a,num, p;
314.     printf("\nEnter two numbers to find the power \n");
315.     printf("number: ");

```

```

316.     scanf("%lf",&a);
317.
318.     printf("power : ");
319.     scanf("%lf",&num);
320.
321.     p=pow(a,num);
322.
323.     printf("\n%lf to the power %lf = %lf \n",a,num,p);
324. }
325. int factorial()
326. {
327.     int i,fact=1,num;
328.     printf("\nEnter a number to find factorial : ");
329.     scanf("%d",&num);
330.     if (num<0){
331.         printf("\nPlease enter a positive number to");
332.         printf(" find factorial and try again. \n");
333.         printf("\nFactorial can't be found for negative");
334.         printf(" values. It can be only positive or 0 \n");
335.         return 1;
336.     }
337.     for(i=1;i<=num;i++)
338.         fact=fact*i;
339.     printf("\n");
340.     printf("Factorial of entered number %d is:%d\n",num,fact);
341.     return 0;
342. }
343.
344. void square(){
345.     printf("\n * * * * * * * * \n");
346.     printf("*      a      *\n");
347.     printf("*          *\n");
348.     printf("*          *\n");
349.     printf("* a          a *\n");
350.     printf("*          *\n");
351.     printf("*          *\n");
352.     printf("*      a      *\n");
353.     printf(" * * * * * * * * \n\n");
354.     printf("\nEnter value for Side of the Square: ");
355.     scanf("%d", &ch.choice_int);
356.     int area = pow(ch.choice_int,2);
357.     printf("\nThe area of the Square is %d\n\n", area);
358. }
359. void rectangle(){
360.     printf("\n * * * * * * * * * * \n");
361.     printf("*          L          *\n");

```

```

362.     printf("*                *\n");
363.     printf("*                *\n");
364.     printf("* W                W *\n");
365.     printf("*                *\n");
366.     printf("*                *\n");
367.     printf("*                L                *\n");
368.     printf(" * * * * * * * * * * \n");
369.     printf("\nEnter values of Length(L) and Weidth(W) Rectangle: \n");
370.     printf("L = ");
371.     scanf("%d", &ch.choice_int);
372.     int l = ch.choice_int;
373.     printf("W = ");
374.     scanf("%d", &ch.choice_int);
375.     int w = ch.choice_int;
376.     int area = l * w;
377.     printf("\nThe area of the Rectangle is %d\n\n", area);
378. }

379. void triangle(){
380.     printf("\n                *\n");
381.     printf("                *|\n");
382.     printf("                * | *\n");
383.     printf("                * | *\n");
384.     printf("                * |H *\n");
385.     printf("                * || *\n");
386.     printf("                * |H *\n");
387.     printf("                * |e *\n");
388.     printf("                * |i *\n");
389.     printf("                * |g *\n");
390.     printf("                * |h *\n");
391.     printf("                * |t *\n");
392.     printf("                | *\n");
393.     printf(" * * * * * * * * * * *\n");
394.     printf("                B(Base)                ");
395.     printf("\nEnter values for Height(H) and Base(B) of the Triangle: \n");
396.     printf("H (Height) = ");
397.     scanf("%f", &ch.choice_float);
398.     float h = ch.choice_float;
399.     printf("B (Base) = ");
400.     scanf("%f", &ch.choice_float);
401.     float b = ch.choice_float;
402.     float area = 0.5 * h * b;
403.     printf("\nThe area of the Triangle is %.3f\n\n", area);
404. }

405. void circle(){
406.     printf("\nEnter value for Radius of the circle: \n");
407.     printf("Radius: ");
408.     scanf("%f", &ch.choice_float);

```

```

409.     float area = 3.1416 * (pow(ch.choice_float, 2));
410.     printf("\nThe area of the Circle is %.3f\n\n", area);
411. }
412. void trapezoid(){
413.     printf("\nEnter values of Base1, Base2 and Vertical Height(H) of the Trapazoid: \n");
414.     printf("Base 1 = ");
415.     scanf("%f", &ch.choice_float);
416.     float b1 = ch.choice_float;
417.     printf("Base2 = ");
418.     scanf("%f", &ch.choice_float);
419.     float b2 = ch.choice_float;
420.     printf("H (Vertical Height) = ");
421.     scanf("%f", &ch.choice_float);
422.     float h = ch.choice_float;
423.     float area = 0.5 * (b1+b2) * h ;
424.     printf("\nThe area of the Triangle is %.3f\n\n", area);
425. }
426. void ellipse(){
427.     printf("\nEnter values for major axis and minor axis redius: \n");
428.     printf("Radius for major axis = ");
429.     scanf("%f", &ch.choice_float);
430.     float major = ch.choice_float;
431.     printf("Radius for minor axis = ");
432.     scanf("%f", &ch.choice_float);
433.     float minor = ch.choice_float;
434.     float area = 3.1416 * major * minor;
435.     printf("\nThe area of the Ellipse is %.3f\n\n", area);
436. }
437.
438. void cylinder_volume(){
439.     printf("\nEnter values of Radius of the Circular base(R) and Height(H) Rectangle: \n");
440.     printf("R = ");
441.     scanf("%f", &ch.choice_float);
442.     float r = ch.choice_float;
443.     printf("H = ");
444.     scanf("%f", &ch.choice_float);
445.     float h = ch.choice_float;
446.     float volume = 3.1416*(pow(r,2))*h;
447.     printf("\nThe volume of the Rectangle is %.3f\n\n", volume);
448. }
449. void cube_volume(){
450.     printf("\nEnter value for Side of the Cube: ");
451.     scanf("%d", &ch.choice_int);
452.     int area = pow(ch.choice_int,3);
453.     printf("\nThe volume of the Cube is %d\n\n", area);
454. }

```

```

455. void cuboid_volume(){
456.     printf("\nEnter values for Length(L), Width(W) and Height(H) of the Trapazoid: \n");
457.     printf("L = ");
458.     scanf("%f", &ch.choice_float);
459.     float l = ch.choice_float;
460.     printf("W = ");
461.     scanf("%f", &ch.choice_float);
462.     float w = ch.choice_float;
463.     printf("H = ");
464.     scanf("%f", &ch.choice_float);
465.     float h = ch.choice_float;
466.     float area = l * w * h ;
467.     printf("\nThe volume of the Rectangular Solid or Cuboid is %.3f\n\n", area);
468. }
469. // void square_volume();
470. // void sphere_volume();
471.

472. void mbr_women(){
473.     printf("\nFor calculating BMR you need to give some value: \n1.Weight(kg) = ");
474.     scanf("%f", &ch.choice_float);
475.     float w = ch.choice_float;
476.     printf("\nHeight(cm) = ");
477.     scanf("%f", &ch.choice_float);
478.     float h = ch.choice_float;
479.     printf("Age(years) = ");
480.     scanf("%f", &ch.choice_float);
481.     float age = ch.choice_float;
482.     float bmr = 655.1 + (9.563 * w) + (1.850 * h) - (4.676 * age);
483.     printf("BMR = %.3f", bmr);
484. }
485. void mbr_men(){
486.     printf("\nFor calculating BMR you need to give some value: \n1.Weight(kg) = ");
487.     scanf("%f", &ch.choice_float);
488.     float w = ch.choice_float;
489.     printf("\nHeight(cm) = ");
490.     scanf("%f", &ch.choice_float);
491.     float h = ch.choice_float;
492.     printf("Age(years) = ");
493.     scanf("%f", &ch.choice_float);
494.     float age = ch.choice_float;
495.     float bmr = 66.5 + (13.75 * w) + (5.003 * h) - (6.75 * age);
496.     printf("BMR = %.3f", bmr);
497. }
498. void cal_men(){
499.     printf("\nFor calculating BMR you need to give some value: \n1.Weight(kg) = ");
500.     scanf("%f", &ch.choice_float);

```



```

501.     float w = ch.choice_float;
502.     printf("\nHeight(cm) = ");
503.     scanf("%f", &ch.choice_float);
504.     float h = ch.choice_float;
505.     printf("Age(years) = ");
506.     scanf("%f", &ch.choice_float);
507.     float age = ch.choice_float;
508.     float bmr = 66.5 + (13.75 * w) + (5.003 * h) - (6.75 * age);
509.     printf("BMR = %.3f\n", bmr);
510.     printf("\n\nChoose your activity factor: \n");
511.     printf("\n1. Light/No Exercise\n2. Light Excercise/Sports 1-3 Days per week\n3. Moderate
Exercise/Sports 3-5 Days per week\n4. Hard exercise/Sports 6-7 Days a week\n5. Very hard
exercise/Sports & a Physical job\n\n");
512.     scanf("%d", &ch.choice_int);
513.     printf("\n");
514.     switch (ch.choice_int){
515.     case 1:
516.         printf("Callorie you need: \nBMR * 1.2 = %.3f", bmr*1.2);
517.         break;
518.     case 2:
519.         printf("Callorie you need: \nBMR * 1.375 = %.3f", bmr*1.375);
520.         break;
521.     case 3:
522.         printf("Callorie you need: \nBMR * 1.55 = %.3f", bmr*1.55);
523.         break;
524.     case 4:
525.         printf("Callorie you need: \nBMR * 1.725 = %.3f", bmr*1.725);
526.         break;
527.     case 5:
528.         printf("Callorie you need: \nBMR * 1.9 = %.3f", bmr*1.9);
529.         break;
530.
531.     default:
532.         printf("INVALID INPUT!! PLEASE TRY TO GIVE INPUT FROM 1-5");
533.         cal_men();
534.         break;
535.     }
536. }
537. void cal_women(){
538.     printf("\nFor calculating BMR you need to give some value: \n1.Weight(kg) = ");
539.     scanf("%f", &ch.choice_float);
540.     float w = ch.choice_float;
541.     printf("\nHeight(cm) = ");
542.     scanf("%f", &ch.choice_float);
543.     float h = ch.choice_float;
544.     printf("Age(years) = ");
545.     scanf("%f", &ch.choice_float);

```

```

546.     float age = ch.choice_float;
547.     float bmr = 655.1 + (9.563 * w) + (1.850 * h) - (4.676 * age);
548.     printf("BMR = %.3f\n", bmr);
549.     printf("\nChoose your activity factor: \n");
550.     printf("\n1. Light/No Exercise\n2. Light Excercise/Sports 1-3 Days per week\n3. Moderate
Exercise/Sports 3-5 Days per week\n4. Hard exercise/Sports 6-7 Days a week\n5. Very hard
exercise/Sports & a Physical job\n\n");
551.     scanf("%d", &ch.choice_int);
552.     printf("\n");
553.     switch (ch.choice_int){
554.     case 1:
555.         printf("Callorie you need: \nBMR * 1.2 = %.3f", bmr*1.2);
556.         break;
557.     case 2:
558.         printf("Callorie you need: \nBMR * 1.375 = %.3f", bmr*1.375);
559.         break;
560.     case 3:
561.         printf("Callorie you need: \nBMR * 1.55 = %.3f", bmr*1.55);
562.         break;
563.     case 4:
564.         printf("Callorie you need: \nBMR * 1.725 = %.3f", bmr*1.725);
565.         break;
566.     case 5:
567.         printf("Callorie you need: \nBMR * 1.9 = %.3f", bmr*1.9);
568.         break;
569.     default:
570.         printf("INVALID INPUT!! PLEASE TRY TO GIVE INPUT FROM 1-5");
571.         cal_women();
572.         break;
573.     }
574. }
575. void cal_need(){
576.     printf("Choose your Gender: \n1. Male\n2. Female\n\n");
577.     scanf("%d", &ch.choice_int);
578.     if (ch.choice_int == 1){
579.         cal_men();
580.     }else if(ch.choice_int == 2){
581.         cal_women();
582.     }else{
583.         cal_need();
584.     }
585.
586. }
587. void dose_cal(){
588.     printf("Choose your Gender: \n1. Male\n2. Female\n\n");
589.     scanf("%d", &ch.choice_int);
590.     if (ch.choice_int == 1){

```

```

591.     cal_men();
592. }else if(ch.choice_int == 2){
593.     cal_women();
594. }else{
595.     cal_need();
596. }
597. }
598.

599. void play_math(){
600.     printf("\nChoose one from given options: \n");
601.     printf("1. Prime Number Detection\n");
602.     printf("2. Swap 2 numbers within their Variables\n");
603.     printf("3. Fibonacci Series\n");
604.     printf("4. Palindrome Number Detection\n");
605.     printf("5. Leap year Detection \n");
606.     printf("6. Armstrong Number Detection\n");
607.     printf("7. Reverse any number \n");
608.     printf("8. Find GCD & LCM\n\n");
609.     scanf("%d", &ch.choice_int);
610.     switch (ch.choice_int){
611.     case 1:prime_num();
612.         break;
613.     case 2:swap();
614.         break;
615.     case 3:fibonacci();
616.         break;
617.     case 4:palindrome();
618.         break;
619.     case 5:leap_year();
620.         break;
621.     case 6:armstrong_num();
622.         break;
623.     case 7:reverse_num();
624.         break;
625.     case 8:gcd_lcm();
626.         break;
627.     default:
628.         play_math();
629.         break;
630.     }
631. }
632. void prime_num(){
633.     int i, num, p = 0;
634.     printf("Please enter a number: \n");
635.     scanf("%d", &ch.choice_int);
636.     num = ch.choice_int;

```

```

637.     for(i=1; i<=num; i++){
638.         if(num%i==0){
639.             p++;
640.         }
641.     }
642.     if(p==2){
643.         printf("Entered number is %d "\
644.             "and it is a prime number.",num);
645.     }else{
646.         printf("Entered number is %d "\
647.             "and it is not a prime number.",num);
648.     }
649. }

650. void swap(){
651.     int A, B;
652.     printf("Please enter the 1st number : ");
653.     scanf("%d",&A);
654.     printf("\nPlease enter the 2nd number : ");
655.     scanf("%d",&B);
656.     printf("\nBefore swapping:\n");
657.     printf("A - %d \nB - %d", A, B);
658.     A = A - B;
659.     B = A + B;
660.     A = B - A;
661.     printf("\nAfter swapping:\n");
662.     printf("A - %d \nB - %d", A, B);
663. }

664. void fibonacci(){
665.     int f1=0, f2=1, fib_ser, cnt=2, lmt;
666.     printf("Please enter the limit of the Fibonacci series :");
667.     scanf("%d",&ch.choice_int);
668.     lmt = ch.choice_int;
669.     printf("\nFibonacci series is: \n%d \n%d \n",f1,f2);
670.     while (cnt < lmt){
671.         fib_ser=f1+f2;
672.         cnt++;
673.         printf("%d\n",fib_ser);
674.         f1=f2;
675.         f2=fib_ser;
676.     }
677. }

678. void palindrome(){
679.     int number, t, rev=0, rmndr;
680.     printf("Please enter a number to check Palindrome : ");
681.     scanf("%d",&ch.choice_int);
682.     printf("\nEntered number: %d", ch.choice_int);
683.     number = ch.choice_int;

```

```

684.     t = ch.choice_int;
685.     while (number > 0){
686.         rmndr = number%10;
687.         rev = rev*10 + rmndr;
688.         number = number/10;
689.     }
690.     printf("\nReversed number: %d", rev);
691.     if(t == rev){
692.         printf("\nEntered number %d is a palindrome", t);
693.     }else{
694.         printf("\nEntered number %d is not a palindrome", t);
695.     }
696. }
697. void leap_year(){
698.     printf("\nPlease enter a year to check whether it is a leap year or not");
699.     scanf("%d", &ch.choice_int);
700.     int year =ch.choice_int;
701.     if ( year%400 == 0)
702.         printf("\n%d is a leap year", year);
703.     else if ( year%100 == 0)
704.         printf("\n%d is not a leap year", year);
705.     else if ( year%4 == 0 )
706.         printf("\n%d is a leap year", year);
707.     else
708.         printf("\n%d is not a leap year", year);
709. }
710. void armstrong_num(){
711.     int num, originalNum, remainder, result = 0;
712.     printf("Enter a three-digit integer: ");
713.     scanf("%d", &ch.choice_int);
714.     originalNum = ch.choice_int;
715.     while (originalNum != 0) {
716.         remainder = originalNum % 10;
717.         result += remainder * remainder * remainder;
718.         originalNum /= 10;
719.     }
720.     if (result == ch.choice_int)
721.         printf("%d is an Armstrong number.", ch.choice_int);
722.     else
723.         printf("%d is not an Armstrong number.", ch.choice_int);
724.
725. }
726. void reverse_num(){
727.     int n, reverse = 0, remainder;
728.     printf("Enter an integer: ");
729.     scanf("%d", &ch.choice_int);
730.     n = ch.choice_int;

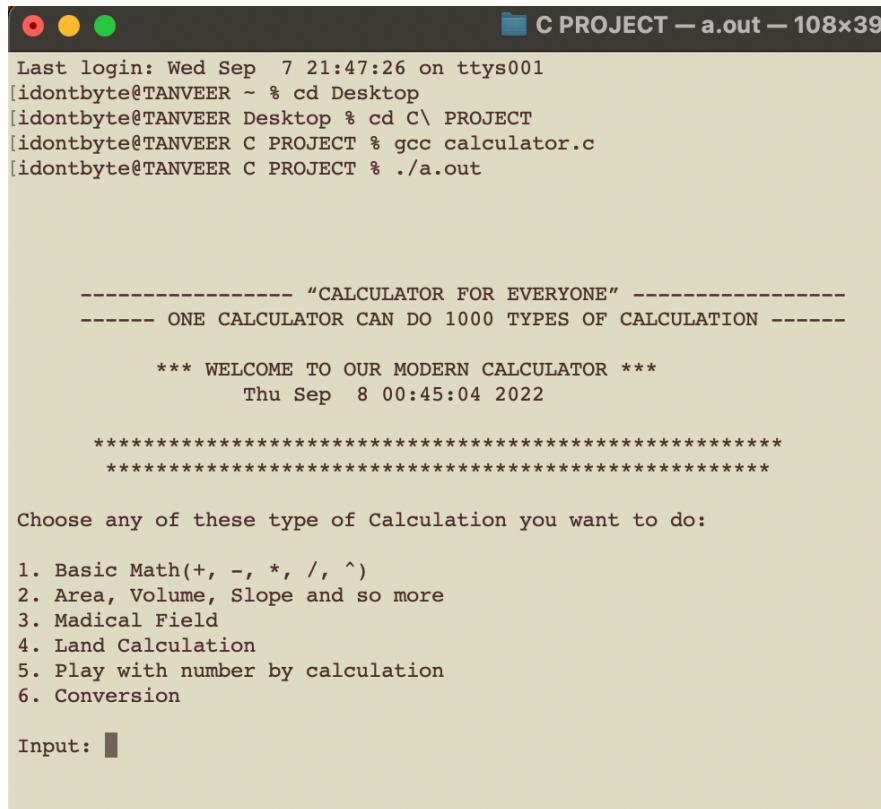
```

```

731.     while (n != 0) {
732.         remainder = n % 10;
733.         reverse = reverse * 10 + remainder;
734.         n /= 10;
735.     }
736.     printf("Reversed number = %d", reverse);
737. }
738. void gcd_lcm(){
739.     int num1, num2, gcd, lcm, remainder, numerator, denominator;
740.     printf("Enter two numbers\n");
741.     scanf("%d %d", &num1, &num2);
742.     if (num1 > num2){
743.         numerator = num1;
744.         denominator = num2;
745.     }else{
746.         numerator = num2;
747.         denominator = num1;
748.     }
749.     remainder = numerator % denominator;
750.     while (remainder != 0){
751.         numerator = denominator;
752.         denominator = remainder;
753.         remainder = numerator % denominator;
754.     }
755.     gcd = denominator;
756.     lcm = num1 * num2 / gcd;
757.     printf("GCD of %d and %d = %d\n", num1, num2, gcd);
758.     printf("LCM of %d and %d = %d\n", num1, num2, lcm);
759. }
760. void c_f(){
761.     float celsius, fahrenheit;
762.     printf("Enter temperature in Celsius: ");
763.     scanf("%f", &celsius);
764.     fahrenheit = (celsius * 9 / 5) + 32;
765.
766.     printf("%.2f Celsius = %.2f Fahrenheit", celsius, fahrenheit);
767. }
768.

```

Screenshots



```
C PROJECT - a.out - 108x39
Last login: Wed Sep  7 21:47:26 on ttys001
[idontbyte@TANVEER ~ % cd Desktop
[idontbyte@TANVEER Desktop % cd C\ PROJECT
[idontbyte@TANVEER C PROJECT % gcc calculator.c
[idontbyte@TANVEER C PROJECT % ./a.out

----- "CALCULATOR FOR EVERYONE" -----
----- ONE CALCULATOR CAN DO 1000 TYPES OF CALCULATION -----

*** WELCOME TO OUR MODERN CALCULATOR ***
Thu Sep  8 00:45:04 2022

*****
*****

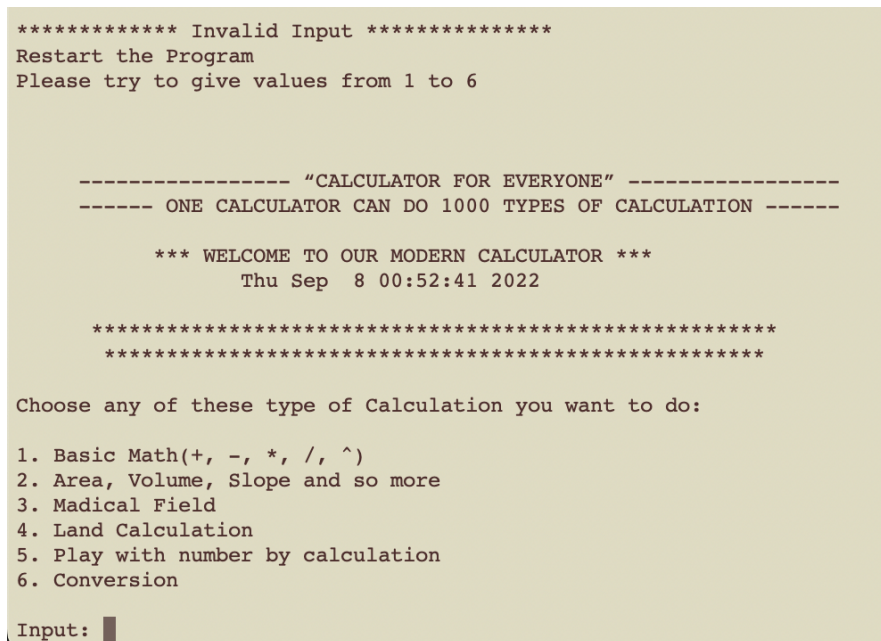
Choose any of these type of Calculation you want to do:

1. Basic Math(+, -, *, /, ^)
2. Area, Volume, Slope and so more
3. Madical Field
4. Land Calculation
5. Play with number by calculation
6. Conversion

Input: █
```

Figure 2: Main Menu interface

When the program is executed, the user will be directed to the main menu interface. The program is introduced with a few lines of texts. Then six selections are made for the user as the user can choose.



```
***** Invalid Input *****
Restart the Program
Please try to give values from 1 to 6

----- "CALCULATOR FOR EVERYONE" -----
----- ONE CALCULATOR CAN DO 1000 TYPES OF CALCULATION -----

*** WELCOME TO OUR MODERN CALCULATOR ***
Thu Sep  8 00:52:41 2022

*****
*****

Choose any of these type of Calculation you want to do:

1. Basic Math(+, -, *, /, ^)
2. Area, Volume, Slope and so more
3. Madical Field
4. Land Calculation
5. Play with number by calculation
6. Conversion

Input: █
```

Figure 3: Invalid value entered (main menu)

If the user accidentally enters an invalid input, an interface will be shown to notify the user to choose again and it notify the user again to enter from 1-6.

```
Input: 1
*** CHOOSE ONE FROM THE LIST

1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Modulus
6. Power
7. Factorial

Input: █
```

Figure 4::Basic Math function

```
*** CHOOSE ONE FROM THE LIST

1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Modulus
6. Power
7. Factorial

Input: 6

Enter two numbers to find the power
number: 7 2
power :
7.000000 to the power 2.000000 = 49.000000
```

Figure 5: Power function from Basic math

```
*** CHOSE ONE FROM THE LIST ***

1. Area
2. Volume
3. Slope
4. Angle
5. Circumference

Input:1

1. Square
2. Rectangle
3. Triangle
4. Circle
5. Trapezoid
6. Ellipse
```

Figure 6: From Area function to 1. Square function

The area of the Triangle is 48.000

The area of the Rectangle is 2400

Chapter 3

Conclusion

Learning Outcome

The Calculator for calculating anything is designed for users to calculate their daily life need for anytime, anywhere and anyhow.

From this assignment, I have learnt to implement a few C concepts in the future projects such as functions, switch statement and while statement, arrays and structures in the program.

Future Scope

- We can add a lot of calculating formulas by the user need.
- We can build the program for mobile usable format so that every mobile phone can have it for the

References

- [1] Author Initial. Author Surname, Title. City: Publisher, Year Published, p. Pages Used.
- [2] A. Rezi and M. Allam, “Techniques in array processing by means of transformations”, in Control and Dynamic Systems, Vol. 69, Multidemsional Systems, C. T. Leondes, Ed. San Diego: Academic Press, 1995, pp. 133-180.
- [3] O. B. R. Strimpel, “Computer graphics”, in McGraw-Hill Encyclopedia of Science and Technology, 8th ed., Vol. 4. New York: McGraw-Hill, 1997, pp. 279-283.
- [4] K. Schwalbe, Information Technology Project Management, 3rd ed. Boston: Course Technology, 2004.