POLITEKNIK MALAYSIA
SULTAN MIZAN ZAINAL ABIDIN
DEPARTMENT OF MECHANICAL ENGINEERING

PRACTICAL TASK

WITH DEV C++

DJM 20032 – C PROGRAMMING

FIRST EDITION/10.2021

PRACTICAL TASK

WITH DEV C++

DJM 20032

C PROGRAMMING

FIRST EDITION/10.2021

Managing Editor

En. Mohd Fadrul Rizal Bin Ab. Rahim

Editor

En. Roshaizul Nizam Bin Mohd Sani

Writers

Pn. Habibah Binti Remeli

En. Mohd Zahari Bin Puteh

Designer

En. Abdul Muadz Bin Abdul Rahim

Terbitan Edisi MAC 2021

Hak Cipta Terpelihara. Tiada bahagian daripada terbitan ini boleh diterbitkan semula, disimpan untuk pengeluaran atau ditukarkan ke dalam sebarang bentuk atau dengan sebarang alat, sama ada dengan cara elektronik, gambar dan rakaman serta sebagainya tanpa kebenaran bertulis daripada Jabatan Pendidikan Politeknik dan Kolej Komuniti, Kementerian Pendidikan Malaysia terlebih dahulu.

Diterbitkan oleh:

Politeknik Sultan Mizan Zainal Abidin,

Km 8, Jalan Paka,

23000 Dungun, Terengganu.

PREFACE

We would like to thank our editorial team very warmly. We also have worked extremely hard and with a lot of dedication to make this ebook a success. It was a pleasure working with our teams. Thank you again.

Any information that enables the publisher to correct any errors or submit any materials in future is welcome.

SYNOPSIS

C Programming course provides an introduction to programmed design and development. Student will learn to design, code, debug, test and document well-structured programs based on technical and engineering problem. Topic covered; software development principle, programming language basic, data types, input and output operation, the use of selection, loops, arrays and function structure.

TABLE OF CONTENTS

- INTRODUCTION TO DEV C++
- PRACTICAL TASK 1
- PRACTICAL TASK 2
- PRACTICAL TASK 3
- PRACTICAL TASK 4

C PROGRAMMING

C Programming is one of the most popular programming languages and can be used for a variety of things, including mobile applications, game development, and enterprise software. Knowing C Programming opens a great deal of doors for you as a developer.

DEV C++

In this course, you'll be exposed to fundamental programming concepts using software Dev C++ and start writing programs right away. You'll build several projects to help you practice and test your knowledge.

LET'S GET STARTED!

DJM 20032

C PROGRAMMING

INTRODUCTION TO SOFTWARE DEV C++

EDITION 2021

POLITEKNIK MALAYSIA
SULTAN MIZAN ZAINAL ABIDIN
DEPARTMENT OF MECHANICAL ENGINEERING

SESSION:.....

DJM 20032 – C PROGRAMMING
INTRODUCTION TO SOFTWARE: DEV C++

NAME: _____
MATRIC NO: _____
DATE / DAY : _____
DATE SUBMIT: _____

TICK	PRACTIKAL SKILL & REPORT ASSESSMENT		MARKS (1 - 5)
☐	PROBLEM SOLVING	IDENTIFY PROBLEMS AND DESIGN FLOW CHART	
☐		PROGRAMIMING WRITING	
☐		TESTING & DEBUGGING PROGRAMMING	
☐	SCIENTIFIC SKILLS	SYNTAX	
☐		COMPLETENESS	
☐		CORRECTNESS	
☐	REPORT	RESULTS	
		DISCUSSION	
☐		CONCLUSION	
☐		QUESTIONS	
☐		REFERENCES	
☐	TOTAL MARKS		

LECTURE'S NAME: _____

DATE RETURN: _____ SIGN : _____

CLO 2 / PLO 3:
Construct a high level programming language in solving variety engineering and scientific problems.

THEORY:
Dev-C++ is a full-featured programming environment and compiler for creating software in C++. Included in the Dev-C++ environment are all of the standard features necessary for writing, compiling, debugging, and executing programs written in C. Strictly for the hard-core C++ programmer, Dev-C++ allows to compose all of source code without many of the hand-holding features and expenses included in many of the available programming environments.

LEARNING OUTCOMES:
Student should be able to:-

1. Be familiar with C programming environment (Dev C++)
2. Use output command.
3. Execute and debug simple programme.
4. Use comments in a programme.

Explain what you LIKED, LOVED, and DISLIKED about today's lesson.

PRACTICAL INSTRUCTION:
Dev C++ Software

1. Switch ON computer.
2. Wait until the desktop appear on the screen.
3. Click START at left bottom of the screen.
4. Click All Programs and scroll mouse to Dev C++.
5. Then, scroll mouse to Dev C++ and click it.
6. The screen of Dev C++ will appear as shown in Figure 1.

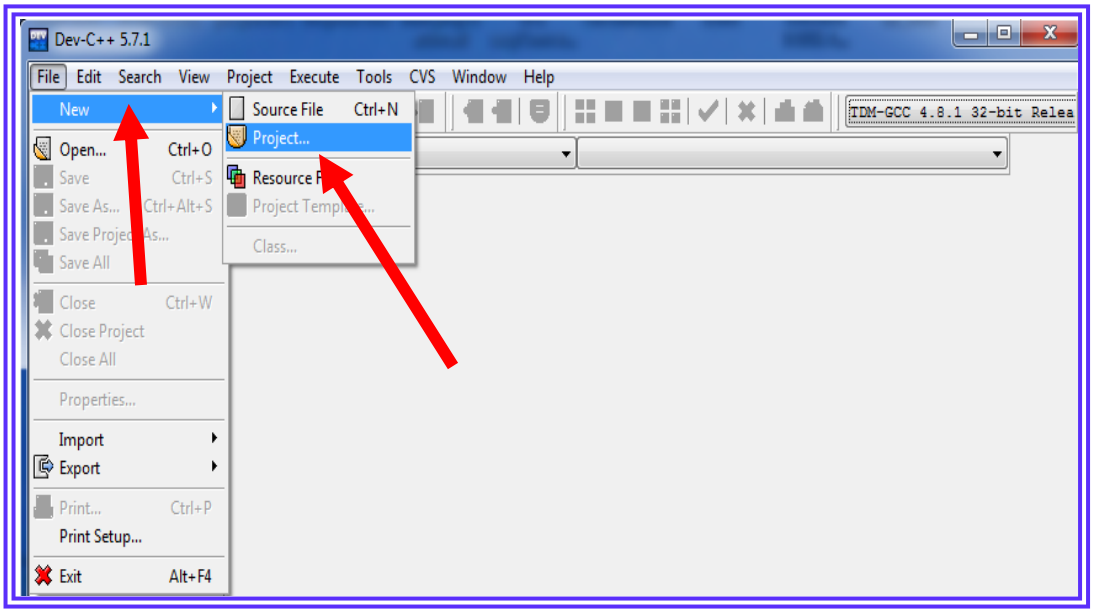


Figure 1

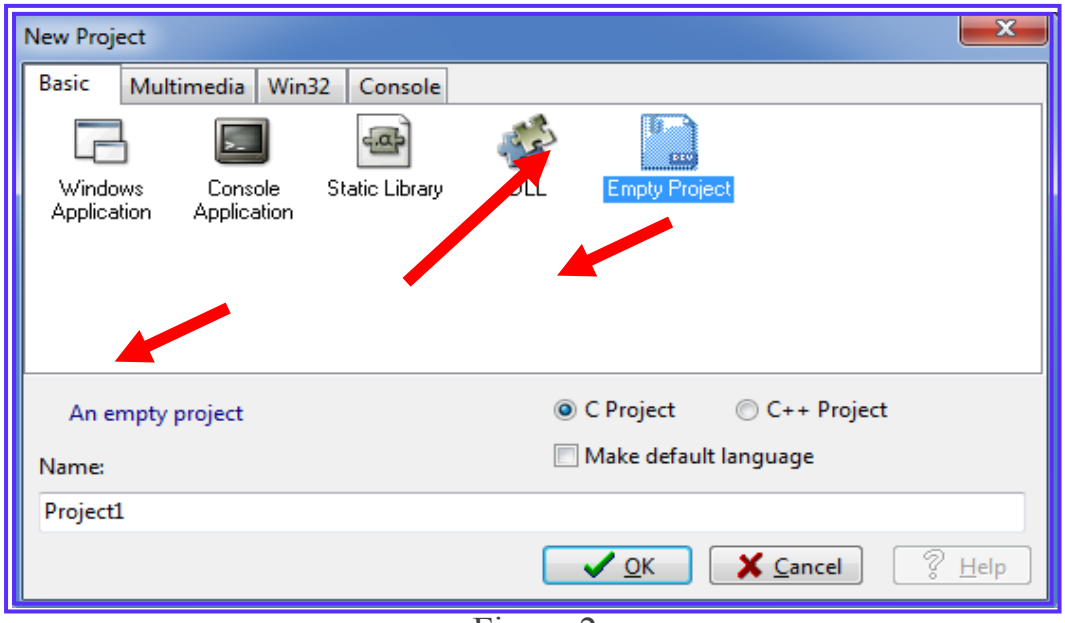


Figure 2

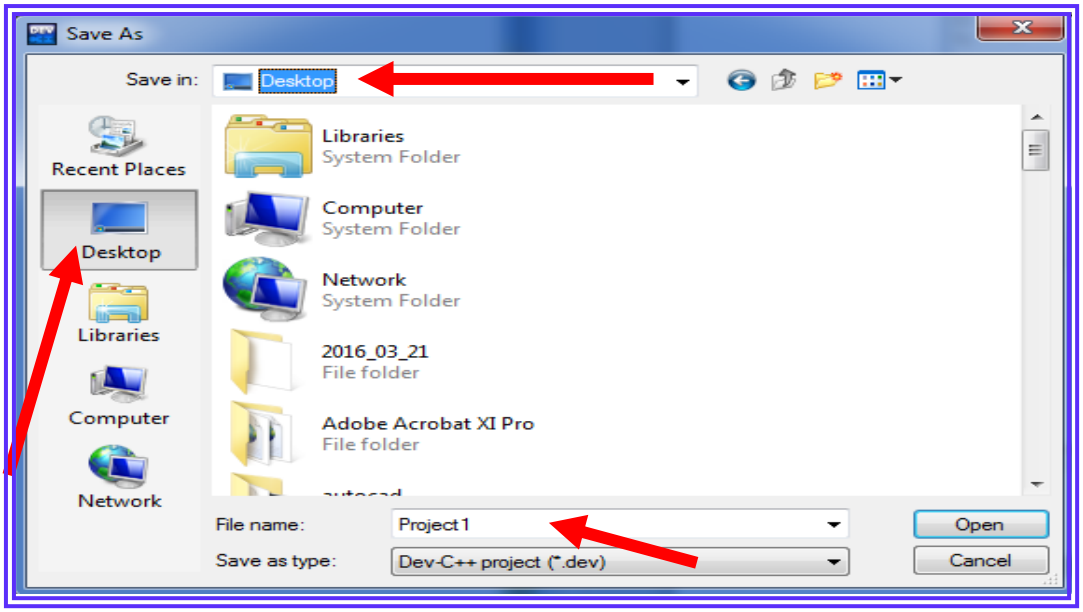


Figure 3

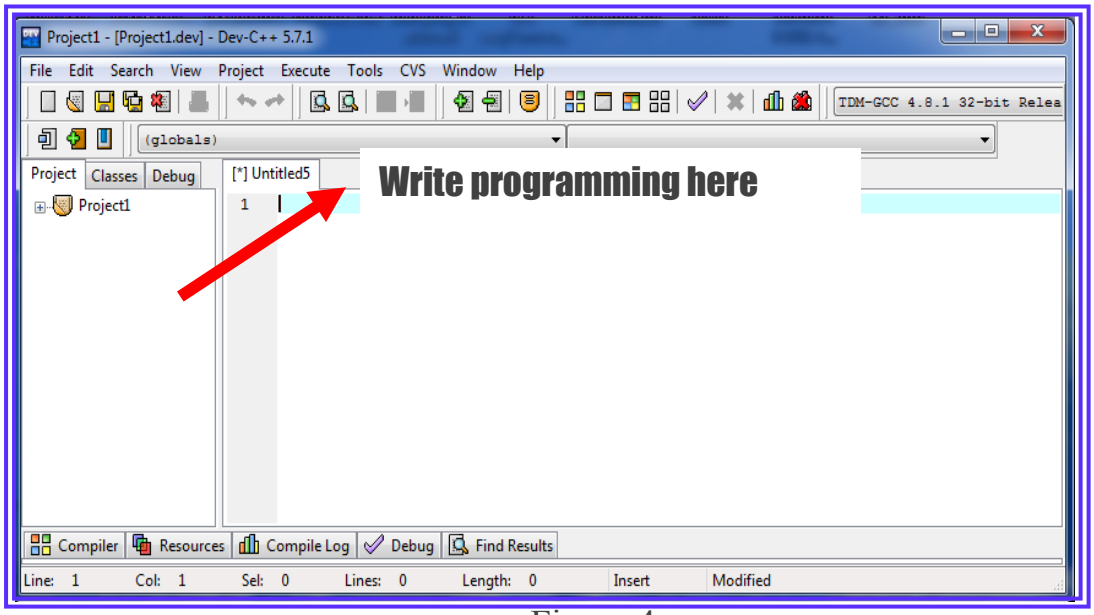


Figure 4

7. Write this program at this window shown in below.

```
#include <stdio.h>
int main( )
{
printf (“Hello World”);
return 0;
}
```

8. After finishing the typing, click Execute and click Compile and run as shown in Figure 5.

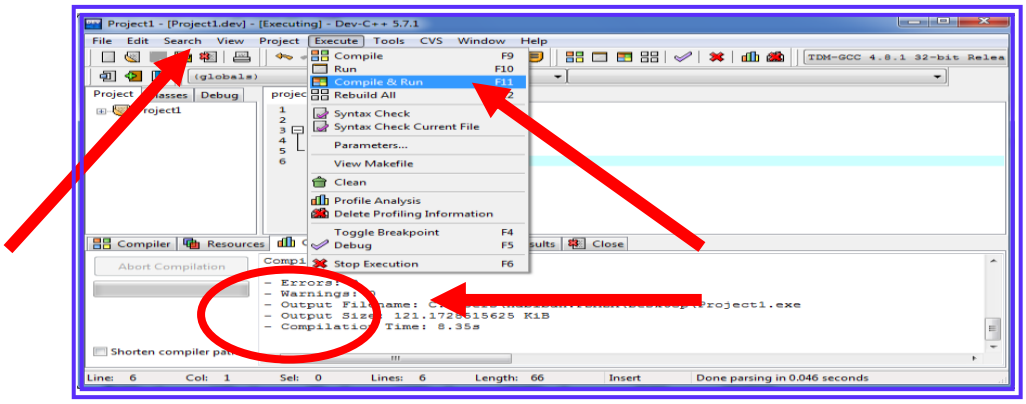


Figure 5

9. Then, the status of compiling window will be shown. If there is error, the correction on the program must be done and repeat compiling.

10. After compiling is finished and error is zero, the output will show as Figure 6.

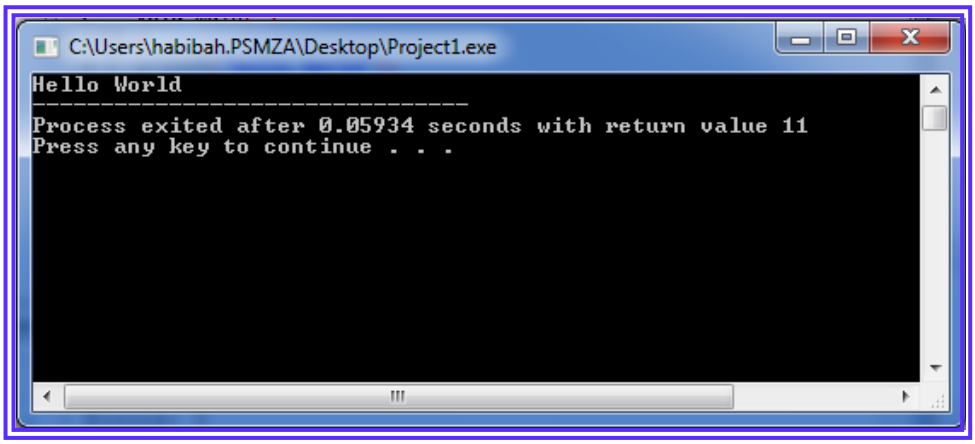


Figure 6 : The output

Students can try these procedures from these programs and write down the output next to the programs. Use Dev C++ that has learned before.

```
//Exercise 1
#include <stdio.h>
main ( )
{
        printf("WhHoo Arre
TranSsformeRrrr?") ;
        printf("\nWEee aArE!") ;

return 0;
}
```

Print Screen Output Exercise 1

(3 marks)

```
//Exercise 2
#include <stdio.h>
main( )
{
printf("SEnyum
S\n\t\tO\n\tK\n\t\tM\n\tO\a\a\a");
return 0;
}
```

Print Screen Output Exercise 2

(3 marks)

```
//Exercise 3
/*To speak well in ENGLISH*/

#include <stdio.h>
main( )
{
        int num=17;                    //declare num
        printf("She SELL sEA SheLL\n") ;
        printf("AT \t\t SEA \n\tSHORE\n") ;
        printf("5768\"n);              //output string
        printf("9999\n");              //output data
        printf("num");                 //output
variables
        return 0;
}
```

Print Screen Output Exercise 3

(3 marks)

```
//Exercise 4
#include <stdio.h>
 int main()
  {
    printf("Name   : Muhammad Yuserwan
Bin Yunus\n");
    printf("\tDOB    : July 14, 2006\n");
    printf("\t\tMobile : 019-9988776655\n");
    return 0;
  }
```

Print Screen Output Exercise 4

(3 marks)

```
//Exercise 5
#include <stdio.h>
 int main()
  {
   printf("    ######\n");
   printf("  ##      ##\n");
   printf(" #\n");
   printf(" #\n");
   printf(" #\n");
   printf(" #\n");
   printf(" #\n");
   printf("  ##      ##\n");
   printf("    ######\n");

 return 0;
 }
```

Print Screen Output Exercise 5

(3 marks)

```
#include <stdio.h>
int main()
{
    int a = 1;
    char b = 'G';
    double c = 3.14;
    float d = 3.5555;

   printf("Hello DEM Student. Let's Learn C Programming Today'!\n");
   printf("\n\n ---Hello! I am an integer / int. My value is %d and my size is %lu  bytes.---\n", a, sizeof(int));
   printf("\n\n ***Hello! I am a character / char. My value is %c and my size is %lu byte.***\n", b, sizeof(char));
   printf("\n\n ^^^Hello! I am a floating / float . My value is %f and my size is %l bytes.^^^\n", c, sizeof(double));
   printf("\n\n <<<Hello! I am a double floating point variable. My value is %lf and my size is %lu bytes.<<<\n",
   c, sizeof(double));
    printf("\t\t\t Bye! See you next class. :)\n");

    return 0;
}
```

Print Screen Output Exercise 6

(3 marks)

TALK is CHEAP SHOW me THE CODE

- BELIEVE IN YOURSELF -

DJM 20032

C PROGRAMMING

PRACTICAL TASK 1

EDITION
2021

DEPARTMENT OF MECHANICAL ENGINEERING

SESSION:.....

PRACTICAL TASK (P) 1: C FUNDAMENTAL SOFTWARE: DEV C++

NAME: _____
MATRIC NO: _____
DATE / DAY : _____
DATE SUBMIT: _____

TICK	PRACTIKAL SKILL & REPORT ASSESSMENT		MARKS (1 - 5)
☐	PROBLEM SOLVING	IDENTIFY PROBLEMS AND DESIGN FLOW CHART	
☐		PROGRAMIMING WRITING	
☐		TESTING & DEBUGGING PROGRAMMING	
☐	SCIENTIFIC SKILLS	SYNTAX	
☐		COMPLETENESS	
☐		CORRECTNESS	
☐	REPORT	RESULTS	
☐		DISCUSSION	
☐		CONCLUSION	
☐		QUESTIONS	
☐		REFERENCES	
☐	TOTAL MARKS		

LECTURE'S NAME: _____

DATE RETURN: _____ SIGN : _____

CLO 2 / PLO 3:
Construct a high level programming language in solving variety engineering and scientific problems.

THEORY:
Dev-C++ is a full-featured programming environment and compiler for creating software in C++. Included in the Dev-C++ environment are all of the standard features necessary for writing, compiling, debugging, and executing programs written in C. Strictly for the hard-core C++ programmer, Dev-C++ allows to compose all of source code without many of the hand-holding features and expenses included in many of the available programming environments.

LEARNING OUTCOMES:
Student should be able to:-

1. Be familiar with C programming environment (Dev C++)
2. Use output command.
3. Execute and debug simple programme.
4. Use assignment operator in a programme.

Explain what you LIKED, LOVED, and DISLIKED about today's lesson.

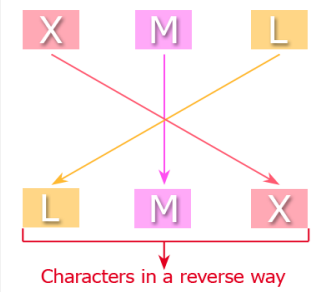
_____

_____

_____

WRITE THE PROGRAMMING GIVEN AND UNDERSTAND FUNDAMENTAL OF C PROGRAMMING.

PROBLEM 1: Write a C program to print X,M,L in a reverse way.



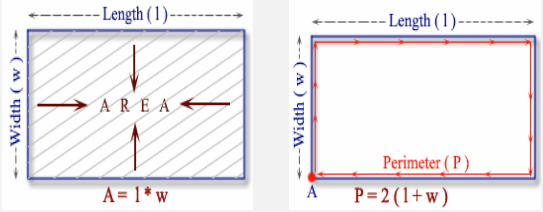
PROGRAMMING SOLUTION:

```
#include <stdio.h>
 int main()
 {
   char char1 = 'X';
   char char2 = 'M';
   char char3 = 'L';

printf("The reverse of %c%c%c is %c%c%c\n",
char1, char2, char3,char3, char2, char1);

return(0);
}
```

Print Screen Problem 1

(3 marks)

PROBLEM 2: Write a C program to compute the perimeter and area of a rectangle with a height of 7 mm and width of 5 mm.



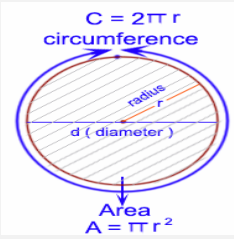
PROGRAMMING SOLUTION:

```
#include <stdio.h>
/* height and width of a rectangle in inches */
int width;
int height;

int area;
int perimeter;

int main() {
        height = 7;
        width = 5;

perimeter = 2*(height + width);
        printf("Perimeter of the rectangle
= %d inches\n", perimeter);
        area = height * width;
        printf("Area of the rectangle = %d
square inches\n", area);
return(0);
}
```

Print Screen Problem 2

(3 marks)

PROBLEM 3: Write a C program to compute the perimeter and area of a circle with a radius of 6 inches.



PROGRAMMING SOLUTION:

*** FIX THE PROGRAMMING FIRST!!!!!!

```
#include <stdio.h>
int main()
{
   int radius;
   float area, perimeter;
   radius = 6;

   perimeter = 2*3.14*radius;
   printf("Perimeter of the Circle = %.3f inches\n",
perimeter);

area = 3.14*radius*radius;
printf("Area of the Circle = %.4f square inches\n", area);

return 0;
}
```

Print Screen Problem 3

(3 marks)

PROBLEM 4: Write a C program to display following variables.

a+ c, x + c,dx + x, ((int) dx) + ax, a + x, s + b, ax + b, s + c, ax + c, ax + ux

Variable declaration :

int a = 125, b = 12345;
long ax = 1234567890;
short s = 4043;
float x = 2.13459;
double dx = 1.1415927;
char c = 'W';
unsigned long ux = 2541567890;

PROGRAMMING SOLUTION:

```
#include <stdio.h>
int main()
{
   int a = 125, b = 12345;
   long ax = 1234567890;
   short s = 4043;
   float x = 2.13459;
   double dx = 1.1415927;
   char c = 'W';
   unsigned long ux = 2541567890;
   printf("a + c =  %d\n", a + c);
   printf("x + c = %f\n", x + c);
   printf("dx + x = %f\n", dx + x);
   printf("((int) dx) + ax =  %ld\n", ((int) dx) + ax);
   printf("a + x = %f\n", a + x);
   printf("s + b =  %d\n", s + b);
   printf("ax + b = %ld\n", ax + b);
   printf("s + c =  %hd\n", s + c);
   printf("ax + c = %ld\n", ax + c);
   printf("ax + ux = %lu\n", ax + ux);
   return 0;
}
```

Print Screen Problem 4

(3 marks)

PROBLEM 5: Write a C program to explain using of operator name.

Operator	Meaning of Operator
+	addition or unary plus
-	subtraction or unary minus
*	multiplication
/	division
%	remainder after division (modulo division)

PROGRAMMING SOLUTION:

```
*** FIX THE PROGRAMMING FIRST!!!!!!
int main()
{
   int a = 7, b = 2, c, d,e,f,g;

   c = a % b;
   d = a / b;
   e = a * b;
   f = a + b;
   g= a - b;

            printf("BAKI %d BAHAGI %d
IALAH: %d\n\n", a,b,c);
            printf("%d BAHAGI %d IALAH: %d\n\n",
a,b,d);
            printf("HASIL DARAB %d DAN %d
IALAH: %d\n\n", a,b,e);
            printf("HASIL TAMBAH %d DAN %d
IALAH:  %d\n\n", a,b,f);
            printf("HASIL TOLAK  %d DAN %d
IALAH: %d\n\n", a,b,g);
```

Print Screen Problem 5

(3 marks)

PROBLEM 6: Write a C program to explain using of assignment operator.

Operator	Example	Same as
=	a = b	a = b
+=	a += b	a = a+b
-=	a -= b	a = a-b
*=	a *= b	a = a*b
/=	a /= b	a = a/b
%=	a %= b	a = a%b

PROGRAMMING SOLUTION:

*** FIX THE PROGRAMMING FIRST!!!!!!

```
// Working of assignment operators
#include <stdio.h>
int main()
{
   int a = 5, c;
   c = a;      // c is 5
   printf("c = %d\n", c);
   c += a;     // c is 10
   printf("c = %d\n", c);
   c -= a;     // c is 5
   printf("c = %d\n", c);pr
   c *= a;     // c is 25
   printf("c = %d\n", c);
   c /= a;     // c is 5
   printf("c = %d\n", c);
   c %= a;     // c = 0
   printf("c = %d\n", c);
   return 0;
}
```

Print Screen Problem 6

(3 marks)

PROBLEM 7: Write a C program to explain using of Increment and Decrement Operators

Operator	Meaning of Operator
++a	increment prefix
a++	increment suffix
--a	decrement prefix
a--	decrement suffix

PROGRAMMING SOLUTION:

```
#include <stdio.h>
int main()
{
    int x=90, y=50;
   printf("\n%d %d",x--,--y);
   printf("\n%d %d",x--,--y);
   printf("\n%d %d",++x,++y);

}
```

Print Screen Problem 7

(3 marks)

PROBLEM 8: Write a C program to explain using of assignment operator.

Operator	**Meaning of Operator**
==	Equal to 5 == 3 is evaluated to 0
>	Greater than 5 > 3 is evaluated to 1
<	Less than 5 < 3 is evaluated to 0
!=	Not equal to 5 != 3 is evaluated to 1
>=	Greater than or equal to 5 >= 3 is evaluated to 1
<=	Less than or equal to 5 <= 3 is evaluated to 0

PROGRAMMING SOLUTION:

```
// Working of relational operators
#include <stdio.h>
int main()
{
   int a = 5, b = 5, c = 10;

   printf("%d == %d is %d \n", a, b, a == b);
   printf("%d == %d is %d \n", a, c, a == c);
   printf("%d > %d is %d \n", a, b, a > b);
   printf("%d > %d is %d \n", a, c, a > c);
   printf("%d < %d is %d \n", a, b, a < b);
   printf("%d < %d is %d \n", a, c, a < c);
   printf("%d != %d is %d \n", a, b, a != b);
   printf("%d != %d is %d \n", a, c, a != c);
   printf("%d >= %d is %d \n", a, b, a >= b);
   printf("%d >= %d is %d \n", a, c, a >= c);
   printf("%d <= %d is %d \n", a, b, a <= b);
   printf("%d <= %d is %d \n", a, c, a <= c);

   return 0;
}
```

Print Screen Problem 8

(3 marks)

PROBLEM 9: Write a C program to explain using of assignment operator.

```
Operators  Meaning of operators
&          Bitwise AND
  12 - 00001100 (In Binary)
  25 - 00011001 (In Binary)

  Bit Operation of 12 and 25
     00001100
  & 00011001
     ________
     00001000   - 8 (In decimal)
|          Bitwise OR
 12 - 00001100 (In Binary)
 25 - 00011001 (In Binary)

 Bitwise OR Operation of 12 and 25
    00001100
  | 00011001
    ________
    00011101   - 29 (In decimal)
^          Bitwise XOR
  12 - 00001100 (In Binary)
  25 - 00011001 (In Binary)

  Bitwise XOR Operation of 12 and 25
     00001100
  ^ 00011001
     ________
     00010101   - 21 (In decimal)
```

PROGRAMMING SOLUTION:

```
#include <stdio.h>
int main()
{
   int a = 12, b = 25;
   printf("\nIN DECIMAL = %d", a&b);// AND
   printf("\nIN DECIMAL = %d", a|b);// OR
   printf("\nIN DECIMAL = %d", a^b); // XOR
   return 0;
}
```

Print Screen Problem 9

(3 marks)

PROBLEM 10: Write a C program to explain using of assignment operator.

Operator	**Meaning of Operator**
++a	increment prefix
a++	increment suffix
--a	decrement prefix
a--	decrement suffix

PROGRAMMING SOLUTION:

```
#include <stdio.h>
int main()
{
   int a = 10, b = 100;
   float c = 10.5, d = 100.5;
   printf("++a = %d \n", ++a);
   printf("--b = %d \n", --b);
   printf("++c = %f \n", ++c);
   printf("--d = %f \n", --d);
   return 0;
}
```

Print Screen Problem 10

(3 marks)

STUDENT SHOULD BE PREPARE BEFORE LAB SESSION (INDEPENDENT LEARNING NON FACE TO FACE).

QUESTIONS: Write the programming to obtain the results AS OUTPUT BELOW AND MAKE SURE YOU MUST USE ASSIGNMENT OPERATOR IN YOUR PROGRAMMING. Given the value of a=5, b=5, e=7, c=13.67, d=10.6. 20 (MARKS)

```
_____________________________ PRACTICAL TASK (P)1_____________________________
5 4
      4 3
            3 2
                  2 1
                        1 0
                              13.67 -1
                                          8 11.6
                                                7 -2
                                                      0 -1
                                                                  8 12.6
_____________ WELL DONE !!!!TULIS NAMA ANDA DAN NO MATRIK DI SINI_____________
_____________ ___________SEE YOU AGAIN FOR THE NEXT PRACTICAL_____________
--------------------------------
```

PROGRAMMING SOLUTION:

(20 marks)

PRINT SCREEN RESULT COMMAND PROMPT

ANSWER THE QUESTIONS BELOW (5 MARKS)

1. Which escape character can be used to begin a new line in C?
a) \a b) \b c) \m d) \n

2. Which escape character can be used to begin a new line in C?
a) \a b) \b c) \m d) \n

3. Which of the following is invalid?
a) ‘’ b) “ “ c) ‘a’ d) ‘abc’

4.Among binary operation which operator represents increment?
a) -- b) ++ c) - d) !

5. Which of the following declarations is not correct?
a) float 3; b) char
c) int b = 4; d) float

(2marks)

DISCUSSION
(5 Marks)

- ----------
- ----------
- ----------
- ----------
- ----------
- ----------

Conclusion
(5 Marks)

- ----------
- ----------
- ----------
- ----------
- ----------
- ----------
- ----------

References
(5 Marks)

- ----------
- ----------
- ----------
- ----------
- ----------
- ----------

CODING

IS TODAY'S LANGUAGE OF CREATIVITY!!

DJM 20032

C PROGRAMMING

PRACTICAL TASK 2

EDITION
2021

DEPARTMENT OF MECHANICAL ENGINEERING

SESSION:.....

DJM 20032 – C PROGRAMMING
PRACTICAL TASK (P) 2: C FUNDAMENTAL
SOFTWARE: DEV C++

NAME: _____
MATRIC NO: _____
DATE / DAY : _____
DATE SUBMIT: _____

TICK	PRACTIKAL SKILL & REPORT ASSESSMENT		MARKS (1 - 5)
□	PROBLEM SOLVING	IDENTIFY PROBLEMS AND DESIGN FLOW CHART	
□		PROGRAMIMING WRITING	
□		TESTING & DEBUGGING PROGRAMMING	
□	SCIENTIFIC SKILLS	SYNTAX	
□		COMPLETENESS	
□		CORRECTNESS	
□	REPORT	RESULTS	
□		DISCUSSION	
□		CONCLUSION	
□		QUESTIONS	
□		REFERENCES	
□	TOTAL MARKS		

LECTURE'S NAME: _____

DATE RETURN: _____ SIGN : _____

CLO 2 / PLO 3:
Construct a high level programming language in solving variety engineering and scientific problems.

THEORY:
Input means to provide the program with some data to be used in the program and Output means to display data on screen or write the data to a printer or a file.

C programming language provides many built-in functions to read any given input and to display data on screen when there is a need to output the result. In this tutorial, we will learn about such functions, which can be used in our program to take input from user and to output the result on screen.

LEARNING OUTCOMES:
Student should be able to:-

1. Be familiar with C programming environment (Dev C++)
2. Use output command.
3. Execute and debug simple programme.
4. Use I/O in a programme.

Explain what you LIKED, LOVED, and DISLIKED about today's lesson.

_____

_____

_____

WRITE THE PROGRAMMING GIVEN AND UNDERSTAND DATA INPUT OUTPUT OF C PROGRAMMING.

PROBLEM 1: Sketch flowchart before Practical Lab session (INDEPENDENT LEARNING NON FACE TO FACE)

PROGRAMMING SOLUTION:

```
#include<stdio.h>
main ()
{
char name [30];
printf ("Enter you name\n");
gets(name);
puts("Welcome to PSMZA DUNGUN");
puts(name);
}
```

Print Screen Output Problem 1

(3 marks)

PROBLEM 2: Sketch flowchart before Practical Lab session (INDEPENDENT LEARNING NON FACE TO FACE)

PROGRAMMING SOLUTION:

```
 #include<stdio.h>
main ()
{
char name [30];
char matrik [20];
printf ("Enter your name\n");
gets(name);
printf ("Enter your matric no:");
gets (matrik);
puts("Welcome to PSMZA DUNGUN");
puts(name);
puts(matrik);
}
```

Print Screen Output Problem 2

(3 marks)

PROBLEM 3: Sketch flowchart before Practical Lab session (INDEPENDENT LEARNING NON FACE TO FACE)

*** FIX THE PROGRAMMING FIRST!!!!!!

```
#include <stdio.h>
 int main()
 {
   int first, second, add, subtract;
   float divide;
    printf("Enter two integers\n");
   scanf("%d%d", &first, &second);
    add       = first + second;
   subtract = first - second;
   multiply = first * second;
   divide     = first / (float)second;
   printf("Sum = %d\n",add);
   printf("Difference = %d\n",subtract);
   printf("Multiplication = %d\n",multiply);
   printf("Division = %.2f\n",divide);
    return 0;
 }
```

Print Screen Output Problem 3

(3 marks)

PROBLEM 4: Sketch flowchart before Practical Lab session (INDEPENDENT LEARNING NON FACE TO FACE)

PROGRAMMING SOLUTION:

*** FIX THE PROGRAMMING FIRST!!!!!!

```
#include<stdio.h>

int main()
{
  int a, b, c;
   printf("Enter two numbers to add\n");
  scanf("%d%d",&a,&b);
   c = a + b;
   printf("Sum of entered numbers = %d\n",c);
   return 0
```

Print Screen Output Problem 4

(3 marks)

PROBLEM 5: Sketch flowchart before Practical Lab session (INDEPENDENT LEARNING NON FACE TO FACE)

PROGRAMMING SOLUTION:

```
#include <stdio.h>
 int main()
{
  int a;
  printf("Enter an integer\n");
  scanf("%d", &a);
   printf("Integer that you have entered is %d\n", a);
   return 0;
}
```

Print Screen Output Problem 5

(3 marks)

PROBLEM 6: Sketch flowchart before Practical Lab session (INDEPENDENT LEARNING NON FACE TO FACE)

PROGRAMMING SOLUTION:

*** FIX THE PROGRAMMING FIRST!!!!!!

```
#include<stdio.h>

{
  int a, b, c, d, x, y, z;
  printf ("\nEnter value for a :  ");
  scanf ("%d",&a);
  printf ("\nEnter value for b :  ");
  scanf ("%d",&b);
  printf ("\nEnter value for c :  ");
  scanf ("%d",&c);
  printf ("\nEnter value for d :  ");
  scanf ("%d", &d);
  x = a + b * c / d - b;
 y = a % b + c * d - a / c;
 z = (a * b * (a + ( c  + d * a / ( c ) )));
  printf ("\nx = %d", x);
  printf ("\ny = %d", y);
  printf ("\nz = %d\n", z);
return 0;
}
```

Print Screen Output Problem 6

(3 marks)

PROBLEM 7: Sketch flowchart before Practical Lab session (INDEPENDENT LEARNING NON FACE TO FACE)

PROGRAMMING SOLUTION:

*** FIX THE PROGRAMMING FIRST!!!!!!

```
#include <stdio.h>
int main()
{
    float num1;
    double num2;
    printf("Enter a number: ");
    scanf("%f", &num1);
    printf("Enter another number: ");
    scanf("%lf", &num);
    printf("num1 = %f\n", num1);
    printf("num2 = %lf", num2);
    return 0;
}
```

Print Screen Output Problem 7

(3 marks)

PROBLEM 8: Sketch flowchart before Practical Lab session (INDEPENDENT LEARNING NON FACE TO FACE)

PROGRAMMING SOLUTION:

*** FIX THE PROGRAMMING FIRST!!!!!!

```
#include <stdio.h>
int main()
{
    float num1;
    double num2;
    printf("Enter a number: ");
    scanf("%f", &num1);
    printf("Enter another number: ");
    scanf("%lf", &num);
    printf("num1 = %f\n", num1);
    printf("num2 = %lf", num2);
    return 0;
}
```

Print Screen Output Problem 8

(3 marks)

PROBLEM 9: Sketch flowchart before Practical Lab session (INDEPENDENT LEARNING NON FACE TO FACE)

PROGRAMMING SOLUTION:

```
#include <stdio.h>
int main()
{
   int a,b,c;
            printf ("Masukkan 3 nombor pilihan anda:");
            scanf("%d %d %d", &a, &b, &c);
   printf("%d + %d - %d",a,b,c);
   return 0;
}
```

Print Screen Output Problem 9

(3 marks)

PROBLEM 10: Sketch flowchart before Practical Lab session (INDEPENDENT LEARNING NON FACE TO FACE)

PROGRAMMING SOLUTION:

```
#include<stdio.h>
int main(){
float number;
printf("enter a number:");
scanf("%f",&number);
printf("cube of number is:%f",number*number*number);
return 0;
}
```

Print Screen Output Problem 10

(3 marks)

QUESTIONS: Write a program in C that reads a name, matric number and courses. Then, Write the program that user can enter 2 numbers and calculate (add, subtract, divide and multiply). **20 (MARKS)**

```
**************************PRACTICAL TASK 2 (P)**************************
Enter YOUR NAME:MUHAMMAD ALI

Enter YOUR MATRIC_NO:13DEM18F2009

Enter YOUR COURSE:DEM

**********add, subtract, divide and multiply between two numbers*********

 Enter two integers
7
5
Sum = 12
Difference = 2
Multiplication = 35
Division = 1.40

**************************WELL DONE!!!!**********************************
**************************SEE YOU FOR NEXT PRACTICAL*********************
```

PROGRAMMING SOLUTION:

(20 marks)

PRINT SCREEN RESULT COMMAND PROMPT

ANSWER THE QUESTIONS BELOW (5 MARKS)

```
#include<stdio.h>
main ()
{
 int x = 24, y = 39  z = 45;
 x = x + y - z;
 y = z + y -2;
 z = z + y + 6;
 print ("\n%d %d %d", x, y, z);
 return 0 ;
 }
```

1. Find and list 2 error programming above.
   ……
   …….....
   (2 MARKS)

2. 2. What will be the output of the following program ?

   x = ……

   y = ……

   z= ……....

   (3 MARKS)

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____(2marks)

DISCUSSION
(5 Marks)

- ----------
- ----------
- ----------
- ----------
- ----------
- ----------

Conclusion
(5 Marks)

- ----------
- ----------
- ----------
- ----------
- ----------
- ----------
- ----------

References
(5 Marks)

- ----------
- ----------
- ----------
- ----------
- ----------
- ----------

CODING is a Programmer's POETRY

DJM 20032

C PROGRAMMING

PRACTICAL TASK 3

EDITION
2021

POLITEKNIK MALAYSIA
SULTAN MIZAN ZAINAL ABIDIN

DEPARTMENT OF MECHANICAL ENGINEERING

SESSION:.....

DJM 20032 – C PROGRAMMING
PRACTICAL TASK (P) 3: C FUNDAMENTAL
SOFTWARE: DEV C++

NAME: _____
MATRIC NO: _____
DATE / DAY : _____
DATE SUBMIT: _____

TICK	PRACTICAL SKILL & REPORT ASSESSMENT		MARKS (1 - 5)
☐	PROBLEM SOLVING	IDENTIFY PROBLEMS AND DESIGN FLOW CHART	
☐		PROGRAMIMING WRITING	
☐		TESTING & DEBUGGING PROGRAMMING	
☐	SCIENTIFIC SKILLS	SYNTAX	
☐		COMPLETENESS	
☐		CORRECTNESS	
☐	REPORT	RESULTS	
☐		DISCUSSION	
☐		CONCLUSION	
☐		QUESTIONS	
☐		REFERENCES	
☐	TOTAL MARKS		

LECTURE'S NAME: _____

DATE RETURN: _____ SIGN : _____

CLO 2 / PLO 3:
Construct a high level programming language in solving variety engineering and scientific problems.

THEORY:
SELECTION STATEMENT: Using decision control statements we can control the flow of program in such a way so that it executes certain statements based on the outcome of a condition (i.e. true or false). We have 3 types of following decision control statements, if..statement, if-else, else-if statement and switch-case statements.

LOOPING STATEMENT: is used for executing a block of statements repeatedly until a given condition returns false. 4 types of Looping statement loops, while loops, do-while loops and break, continue and go to statement.

ARRAY DATA STRUCTURE: An array is a group (or collection) of same data types. For example an int array holds the elements of int types while a float array holds the elements of float types.

Why we need Array in C Programming?
1) Define 100 variables with int data type and then perform 100 scanf() operations to store the entered values in the variables and then at last calculate the average of them.
2) Have a single integer array to store all the values, loop the array to store all the entered values in array and later calculate the average.

LEARNING OUTCOMES:
Student should be able to:-

5. Be familiar with C programming environment (DEV C)
2. Write C programme using selection statements if, if..else and switch..case.
3. Explain the function of selection statements.
4. Provide students with basic programming skills using selection statements.

Explain what you LIKED, LOVED, and DISLIKED about today's lesson.

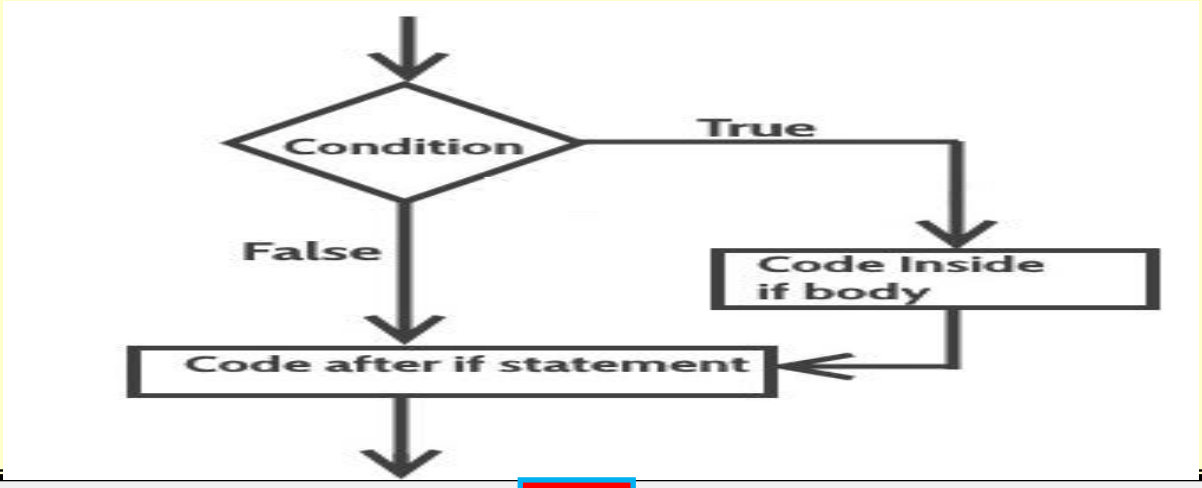
_____

_____

_____

WRITE THE PROGRAMMING GIVEN AND UNDERSTAND CONTROL STATEMENT OF C PROGRAMMING.

PROBLEM 1:
UNDERSTAND THE FLOW CHART AND DISCUSS WITH YOUR FRIENDS BEFORE PRACTICAL LAB SESSION **(INDEPENDENT LEARNING** NON FACE TO FACE)
)



PROGRAMMING SOLUTION:

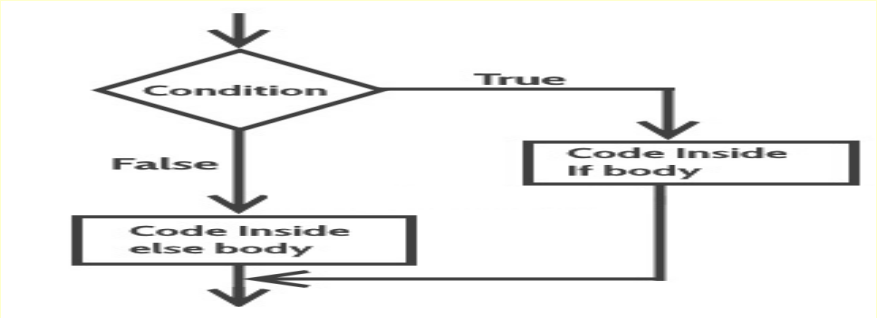
```
#include <stdio.h>
int main()
{
    int x, y;
    printf("enter the value of x:");
    scanf("%d", &x);
    printf("enter the value of y:");
    scanf("%d", &y);
    if (x>y)
    {
        printf("x is greater than y\n");
    }
    if (x<y)
    {
        printf("x is less than y\n");
    }
    if (x==y)
    {
        printf("x is equal to y\n");
    }
    printf("End of Program");
    return 0;
    }
```

Print Screen Output Problem 1

(3 marks)

PROBLEM 2:
UNDERSTAND THE FLOW CHART AND DISCUSS WITH YOUR FRIENDS BEFORE PRACTICAL LAB SESSION **(INDEPENDENT LEARNING** NON FACE TO FACE)

If else statement



PROGRAMMING SOLUTION:

```
#include <stdio.h>
int main()
{
   int age;
   printf("Enter your age:");
   scanf("%d",&age);
   if(age >=18)
printf("You are eligible for voting");
   else
printf("You are not eligible for voting");
   return 0;
}
```

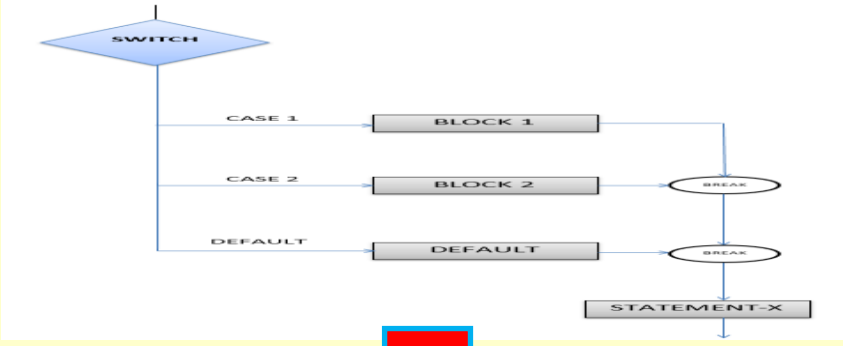
Print Screen Output Problem 2

(3 marks)

PROBLEM 3:

UNDERSTAND THE FLOW CHART AND DISCUSS WITH YOUR FRIENDS BEFORE PRACTICAL LAB SESSION (INDEPENDENT LEARNING NON FACE TO FACE)

Switch ….



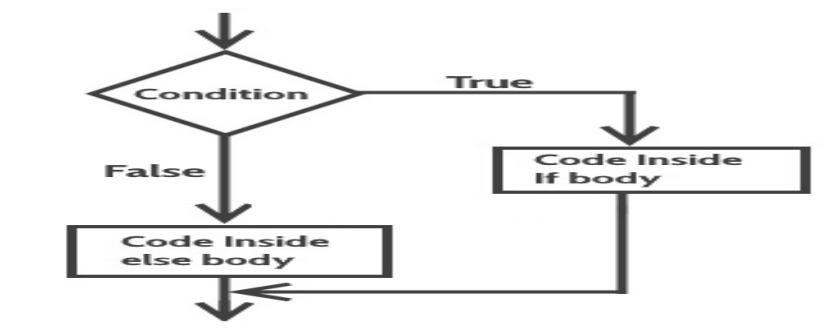
PROGRAMMING SOLUTION: ** FIX THE PROGRAMMING FIRST

```
#include<stdio.h>
#include<conio.h>
int main()
{
   int choice, a, b;
   int sum, div, mul, sub;
   char c= 'y';
   while(c=='y')
   {
      printf("\nAddition  -> 1\nSubtraction  -> 2
\nMultiplication  -> 3 \nDivision  -> 4\n");
      printf("\aEnter your choice of operation to perform -:
\n");
      scanf("%d",& choice);
      printf("\a\nEnter two number :- \n");
      scanf("%d%d",&a,&b);
      switch(choice)
      {
         case 1: sum = a + b;
         printf("\a\n Sum = %d", sum);
         break;

         case 2: sub = a - b;
         printf("\a\n Difference = %d", sub);
         break;

         case 3: mul = a * b;
         printf("\a\n Multiplication = %d", mul);
         break;

         case 4: div = a \ b;
         printf("\a\n Division = %d", div);
         break;

         default : printf("\aWRONG CHOICE");
      }
      printf("\nDo you want to continue [y / n]");
      c=getch();
   }
   getch();
}
```

Print Screen Output Problem 3

(3 marks)

PROBLEM 4:
UNDERSTAND THE FLOW CHART AND DISCUSS WITH YOUR FRIENDS BEFORE PRACTICAL LAB SESSION (INDEPENDENT LEARNING NON FACE TO FACE)
else..if statement



PROGRAMMING SOLUTION:

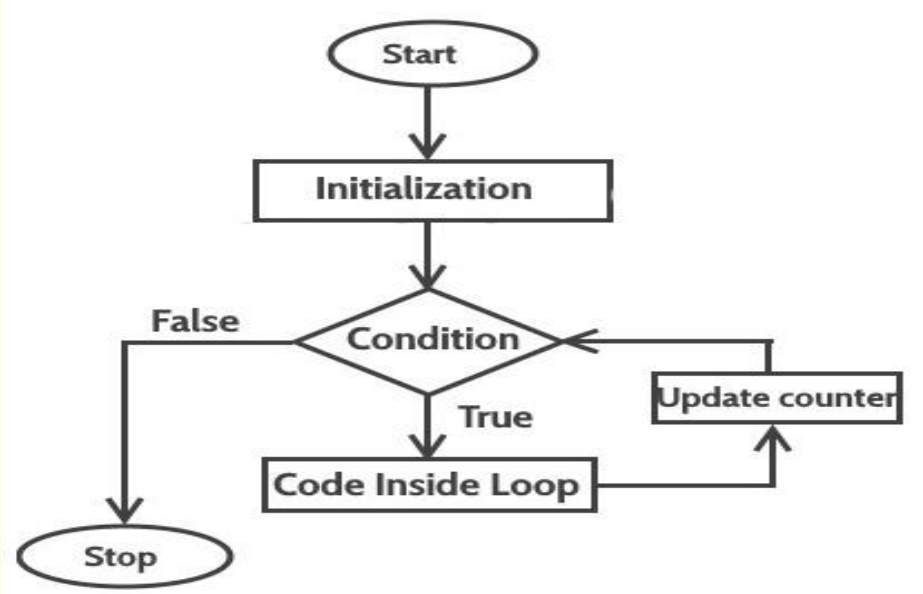
```
#include <stdio.h>
int main()
{
   int number;
   printf("Enter an integer: ");
   scanf("%d", &number);
   // True if the remainder is 0
   if  (number%2 == 0) {
      printf("%d is an even
integer.",number);
   }
   else {
      printf("%d is an odd
integer.",number);
   }
   return 0;
}
```

Print Screen Output Problem 4

(3 marks)

PROBLEM 5:
UNDERSTAND THE FLOW CHART AND DISCUSS WITH YOUR FRIENDS BEFORE PRACTICAL LAB SESSION (INDEPENDENT LEARNING NON FACE TO FACE)

loop



PROGRAMMING SOLUTION:

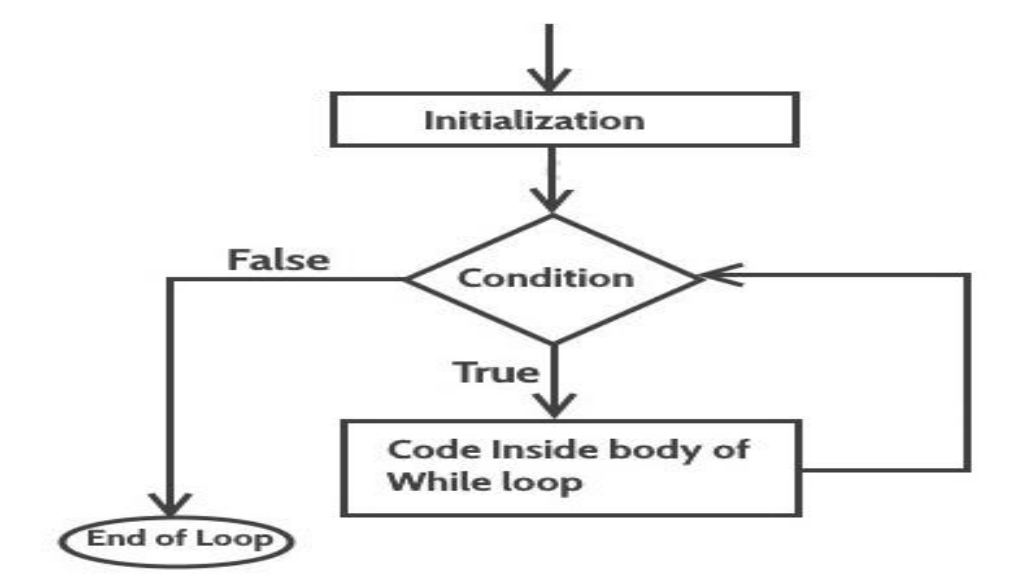
```
#include <stdio.h>
int main()
{
  int i;
  for (i=1; i<=3; i++)
  {
     printf("%d\n", i);
  }
  return 0;
}
```

Print Screen Output Problem 5

(3 marks)

PROBLEM 6:
UNDERSTAND THE FLOW CHART AND DISCUSS WITH YOUR FRIENDS BEFORE PRACTICAL LAB SESSION (INDEPENDENT LEARNING NON FACE TO FACE)

c- while loop



PROGRAMMING SOLUTION:

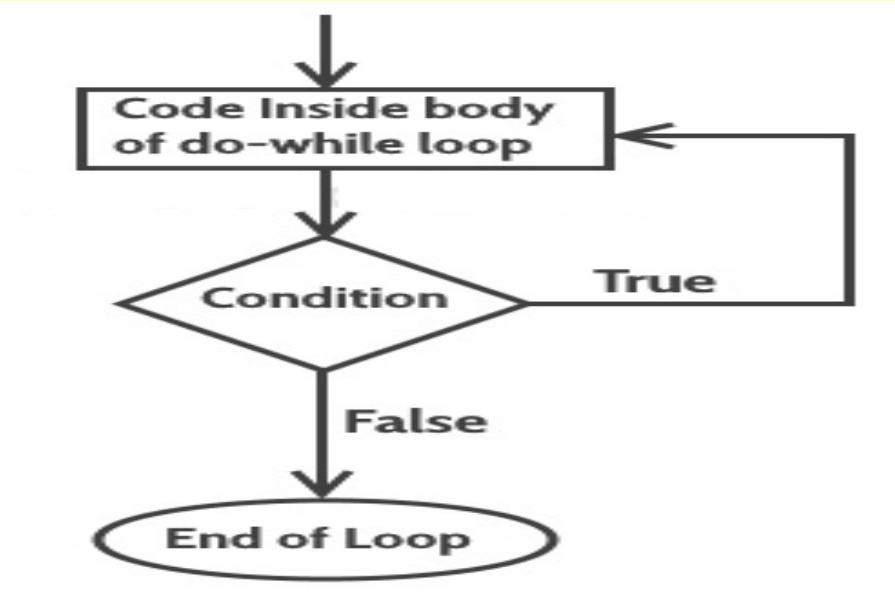
```
#include <stdio.h>
int main()
{
   int i=1, j=1;
   while (i <= 4 || j <= 3)
   {
         printf("%d %d\n",i, j);
         i++;
         j++;
   }
   return 0;
}
```

Print Screen Output Problem 6

(3 marks)

PROBLEM 7:
UNDERSTAND THE FLOW CHART AND DISCUSS WITH YOUR FRIENDS BEFORE PRACTICAL LAB SESSION (INDEPENDENT LEARNING NON FACE TO FACE)

do- while loop



PROGRAMMING SOLUTION:

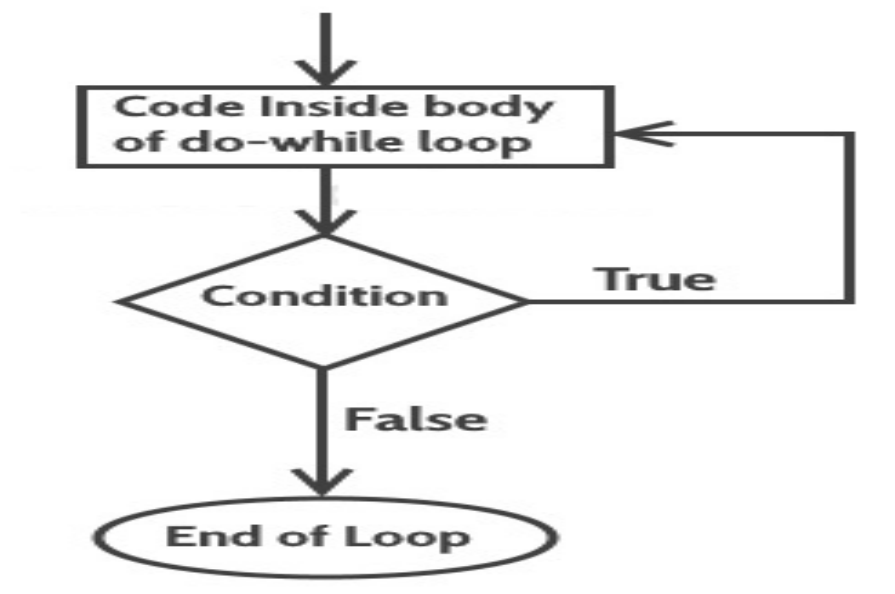
```
#include <stdio.h>
int main()
{
        int j=0;
        do
        {
printf("Value of variable j is: %d\n",
j);
                j++;
        }while (j<=3);
        return 0;
}
```

Print Screen Output Problem 7

(3 marks)

PROBLEM 7:
UNDERSTAND THE FLOW CHART AND DISCUSS WITH YOUR FRIENDS BEFORE PRACTICAL LAB SESSION (INDEPENDENT LEARNING NON FACE TO FACE)

do- while loop



PROGRAMMING SOLUTION:

```
#include <stdio.h>
int main()
{
        int j=0;
        do
        {
printf("Value of variable j is: %d\n",
j);
                j++;
        }while (j<=3);
        return 0;
}
```

Print Screen Output Problem 7

(3 marks)

PROBLEM 8:
UNDERSTAND THE FLOW CHART AND DISCUSS WITH YOUR FRIENDS BEFORE PRACTICAL LAB SESSION (INDEPENDENT LEARNING NON FACE TO FACE)

C array

val[0]	val[1]	val[2]	val[3]	val[4]	val[5]	val[6]
11	22	33	44	55	66	77
88820	88824	88828	88832	88836	88840	88844

All the array elements occupy contigious space in memory. There is a difference of 4 among the addresses of subsequent neighbours, this is because this array is of integer types and an integer holds 4 bytes of memory.

Memory representation of array

PROGRAMMING SOLUTION:

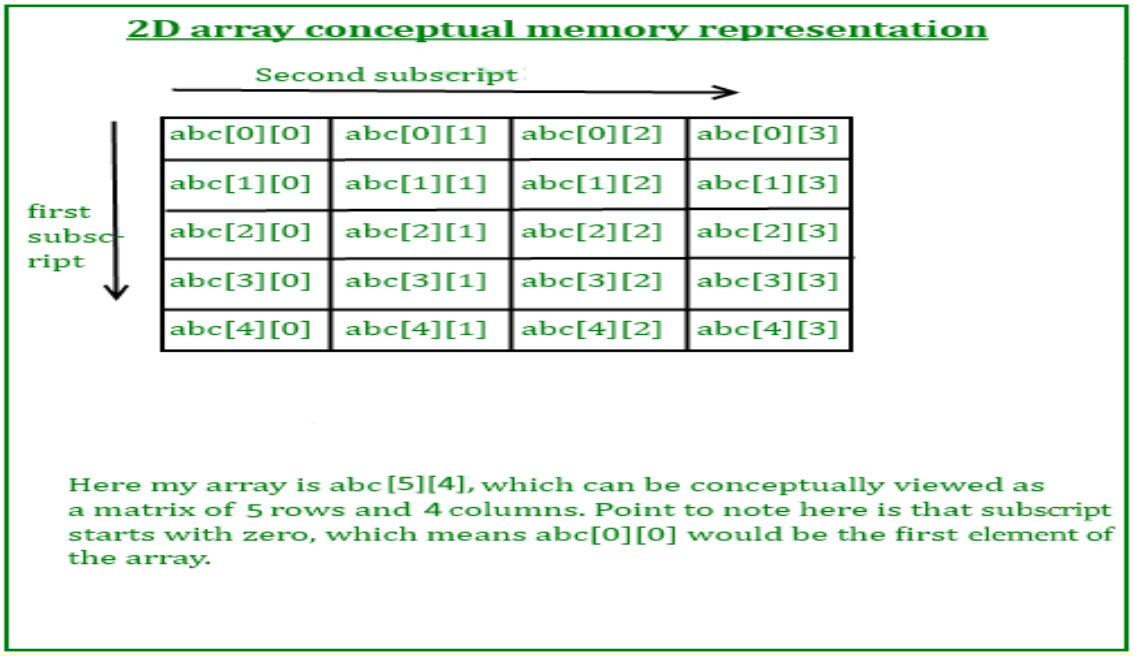
```
#include <stdio.h>
int main()
{
   int avg = 0; int sum =0;  int x=0; int  num[4];
/* Array- declaration – length 4*/
   /* We are using a for loop to traverse through
the array. while storing the entered values in the
array  */
   for (x=0; x<4;x++)
   {
      printf("Enter number %d \n", (x+1));
      scanf("%d", &num[x]);
   }
   for (x=0; x<4;x++)
   {
      sum = sum+num[x];
   }
   avg = sum/4;
   printf("Average of entered number is: %d",
avg);
   return 0;
}
```

Print Screen Output Problem 8

(3 marks)

PROBLEM 9:
UNDERSTAND THE FLOW CHART AND DISCUSS WITH YOUR FRIENDS BEFORE PRACTICAL LAB SESSION (INDEPENDENT LEARNING NON FACE TO FACE)

2 Dimensional Array



PROGRAMMING SOLUTION:

```
#include<stdio.h>
int main(){
    int disp[2][3]; /* 2D array declaration*/
  /*Counter variables for the loop*/
  int i, j;
  for(i=0; i<2; i++) {
    for(j=0;j<3;j++) {
      printf("Enter value for disp[%d][%d]:", i, j);
      scanf("%d", &disp[i][j]);
    }
  }
  //Displaying array elements
  printf("Two Dimensional array elements:\n");
  for(i=0; i<2; i++) {
    for(j=0;j<3;j++) {
      printf("%d ", disp[i][j]);
      if(j==2){
        printf("\n");
      }
    }
  }
  return 0;
}
```

Print Screen Output Problem 9

(3 marks)

QUESTIONS: Write a programming with the output below. Customer can choose the menu, enter the cash and calculate the balance. If the money is not enough to pay, customer will advise to pay at counter. Customers also will be noticed if enter wrong selections.

(20 marks)

```
                                WELCOME TO ROSMAH RESTAURANT
                                        SELF SERVICE
1=Burger
2=French Fries
3=Pizza
4=Sandwiches


Please Enter your order
Enter your choice 1,2,3,4
5
Invalid your choice
```

Customer choose invalid choice

```
                                WELCOME TO ROSMAH RESTAURANT
                                        SELF SERVICE
1=Burger
2=French Fries
3=Pizza
4=Sandwiches

Please Enter your order
Enter your choice 1,2,3,4
2
Your order is French_fries
Please enter your quantity:   3
Your total charges is:RM 16.50

Enter your Payment note:  100

Your balance is RM:83.50

                        ←NJOY YOUR MEAL. PLAESE COME AGAIN. TQ
```

Customer pay and get the balance

```
                                WELCOME TO ROSMAH RESTAURANT
                                        SELF SERVICE
1=Burger
2=French Fries
3=Pizza
4=Sandwiches

Please Enter your order
Enter your choice 1,2,3,4
4
your order is Sandwiches
Please enter your quantity: 22
Your total charges is:RM 33.00

Enter your Payment note:  20
Your money is not enough. Please go to counter number 1.

Your balance is RM:0.00

                        ←NJOY YOUR MEAL. PLAESE COME AGAIN. TQ
```

Customer not enough money to pay

Programming solution

(20 marks)

PRINT SCREEN RESULT

ANSWER THE QUESTIONS BELOW (5 MARKS)

Tick √ or X for the following questions.

1. Selection /decision control structure are if, if…else and switch-case. ( ).
2. Loops are used to repeat statements. ( ).
3. Looping control structure are for, while, do…while. ( ).
4. DO..WHILE loops are useful for things that want to loop at least twice. ( ).
5. If the condition result is true, the control program will execute the instruction within the FALSE loop operation.( ).

(5marks)

Why?

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____(2marks)

DISCUSSION (5 Marks)

- ----------
- ----------
- ----------
- ----------
- ----------
- ----------

Conclusion (5 Marks)

- ----------
- ----------
- ----------
- ----------
- ----------
- ----------
- ----------

References (5 Marks)

- ----------
- ----------
- ----------
- ----------
- ----------
- ----------

```
while (alive) {
    eat();
    sleep();
    code();
    repeat();
}
```

```
if(YOU==BelieveDreams)
{
    if(WORK==HARD)
    {
        DreamsComeTrue();
    }
}
```

LET'S FINISH YOUR LAST PRACTICAL TASK....

DJM 20032

C PROGRAMMING

PRACTICAL TASK 4

EDITION 2021

POLITEKNIK MALAYSIA
SULTAN MIZAN ZAINAL ABIDIN

DEPARTMENT OF MECHANICAL ENGINEERING

SESSION:.....

DJM 20032 – C PROGRAMMING
PRACTICAL TASK (P) 4: C FUNDAMENTAL
SOFTWARE: DEV C++

NAME: _____
MATRIC NO: _____
DATE / DAY : _____
DATE SUBMIT: _____

TICK	PRACTIKAL SKILL & REPORT ASSESSMENT		MARKS (1 - 5)
☐	PROBLEM SOLVING	IDENTIFY PROBLEMS AND DESIGN FLOW CHART	
☐		PROGRAMIMING WRITING	
☐		TESTING & DEBUGGING PROGRAMMING	
☐	SCIENTIFIC SKILLS	SYNTAX	
☐		COMPLETENESS	
☐		CORRECTNESS	
☐	REPORT	RESULTS	
☐		DISCUSSION	
☐		CONCLUSION	
☐		QUESTIONS	
☐		REFERENCES	
☐	TOTAL MARKS		

LECTURE'S NAME: _____

DATE RETURN: _____ SIGN : _____

CLO 2 / PLO 3:
Construct a high level programming language in solving variety engineering and scientific problems.

THEORY:
A function is a named, independent section of C code that performs a specific task and optionally returns a value to the calling program.

2. There are basically TWO (2) types of functions:
-**Predefined functions** - available in the C / C++ standard library such as stdio.h, math.h, string.h etc.
- User-defined functions - As functions are defined by users, they are called user-defined functions.

user-defined functions have contained the block of statements which are written by the user to perform a task

Functions very important in c programming because
-problem can be viewed in a smaller scope.
- Program development are much faster compared to the common structure.
- Program becomes easier to maintain

LEARNING OUTCOMES:
Student should be able to:-
6. Be familiar with C programming environment (Dev C++)
7. Use output command.
8. Execute and debug simple programme.
9. Understanding and using C functions that receive and return value.

Explain what you LIKED, LOVED, and DISLIKED about today's lesson.

WRITE THE PROGRAMMING GIVEN AND UNDERSTAND THE FUNCTION OF C PROGRAMMING.

PROBLEM 1:

```
#include <stdio.h>

   int sum (int, int);//function declaration
   int main (void)
   {
      int total;
printf("\n\n Function : a simple structure of
function :\n");
printf("--------------------------------------\n");

      total = sum (5, 6);//function call
      printf ("The total is :  %d\n", total);
      return 0;
   }

   int sum (int a, int b) //function definition
   {
        int s;
       s=a+b;
      return s; //function returning a value
  }
```

Print Screen Output Problem 1

(3 marks)

PROBLEM 2:

```
#include <stdio.h>
int main()
{       int i, j;

//print one row of number 8
      for(i=1; i<=10; i++)
      printf("8");
       printf("\n");   //go to new line
//print one row of number 9
       for(j=1; j<=10; j++)
       printf("9");
       printf("\n");   //go to new line
       return 0;
```

Print Screen Output Problem 2

(3 marks)

PROBLEM 3:

```
#include <stdio.h>
void display(int);   //function prototype
int main()
{
        display(8);          //function call
        display(9);          //function call
        return 0;
}

void display(int value)     //function
definition
{      int i;
      for(i=1; i<=10; i++)
            printf("%d", value);
      printf("\n");   //go to new line
}
```

Print Screen Output Problem 3

(3 marks)

PROBLEM 4:

```
#include <stdio.h>
int addNumbers(int a, int b);
// function prototype
int main()
{
    int n1,n2,sum;
    printf("Enters two
numbers: ");
    scanf("%d %d",&n1,&n2);
    sum = addNumbers(n1, n2);
// function call
    printf("sum = %d",sum);
    return 0;
}
int addNumbers(int a, int b)
// function definition
{
    int result;
    result = a+b;
    return result;
// return statement
}
```

Print Screen Output Problem 4

(3 marks)

PROBLEM 5:

```
#include<stdio.h>

void swap(int *,int *);
int main()
{
   int n1,n2;
         printf("\n\n Function : swap two
numbers using function :\n");
         printf("-----------------------------------
\n");
   printf("Input 1st number : ");
   scanf("%d",&n1);
   printf("Input 2nd number : ");
   scanf("%d",&n2);

   printf("Before swapping: n1 = %d, n2
= %d ",n1,n2);
         //pass the address of both variables to
the function.
   swap(&n1,&n2);

   printf("\nAfter swapping: n1 = %d, n2
= %d \n\n",n1,n2);
   return 0;
}

void swap(int *p,int *q)
{
         //p=&n1 so p store the address of n1,
so *p store the value of n1
         //q=&n2 so q store the address of n2,
so *q store the value of n2

   int tmp;
   tmp = *p; // tmp store the value of n1
   *p=*q;    // *p store the value of *q that is
value of n2
   *q=tmp;   // *q store the value of tmp that
is the value of n1
}
```

Print Screen Output Problem 5

(3 marks)

PROBLEM 6:

```
#include <stdio.h>

double square(double num)
{
   return (num * num);
}
int main()
{
   int num;
   double n;
printf("\n\n Function : find square of
any number :\n");
printf("-----------------------------------
-\n");

   printf("Input any number for
square : ");
   scanf("%d", &num);
   n = square(num);
   printf("The square of %d
is : %.2f\n", num, n);
   return 0;
}
```

Print Screen Output Problem 6

(3 marks)

PROBLEM 7:

```
#include<stdio.h>
int rect_area(int l,int w)
{
    int area;
    area=l*w;
    return area;
}
int rect_perimeter(int l,int w)
{
    int p;
    p=2*(l+w);
    return p;
}
int main()
{

    float a,b,x,y;
    printf("Enter Length of Rectangle : ");
    scanf("%f",&a);
    printf("\nEnter Width of Rectangle : ");
    scanf("%f",&b);
    x=rect_area(a,b); // calling function
rect_area
    y=rect_perimeter(a,b); // calling function
perimeter
    printf("\nArea of Rectangle
= %f\n\nPerimeter of Rectangle = %f",x,y);
    return 0;
}
```

Print Screen Output Problem 7

(3 marks)

PROBLEM 8:

```
#include<stdio.h>
void tables(int);

int main()
{
    int num;
    printf("Enter a positive number\n");
    scanf("%d", &num);
    printf("\nMultiplication Table
for %d is:\n", num);
   tables(num);
   return 0;
}

void tables(int num)
{
    int count;
      for(count = 1; count <= 12;
count++)
    {
        printf("%d x %d = %d\n", num,
count, num*count);
    }
}
```

Print Screen Output Problem 8

(3 marks)

PROBLEM 9:

```
#include <stdio.h>
/* function return type is void and it doesn't
have parameters*/
void introduction()
{
   printf("Hi\n");
   printf("I'M DAVID 19 YEARS OLD \n");
   printf("NOT COVID 19");
   /* There is no return statement inside this
function, since its
    * return type is void
    */
}
int main()
{
    /*calling function*/
    introduction();
    return 0;
}
```

Print Screen Output Problem 9

(3 marks)

PROBLEM 10:

```
#include <stdio.h>
void salaryGRAB(int  *GAJI_POKOK, int
b)
{
   *GAJI_POKOK = *GAJI_POKOK+b;
}
int main()
{
   int salary=0, bonus=0;
   printf("GRAB FOOD E-SALARY\n\n");
        printf("Enter the employee current
salary:");
   scanf("%d", &salary);
   printf("Enter bonus:");
   scanf("%d", &bonus);
   salaryGRAB(&salary, bonus);
   printf("Final salary: %d", salary);
   return 0;
}
```

Print Screen Output Problem 9

(3 marks)

STUDENT SHOULD BE PREPARE BEFORE LAB SESSION (INDEPENDENT LEARNING NF2F).

QUESTIONS: Write a program below and Compile to run the program. Fix the programming if the output not appears.

```
#include<stdio.h>
#include<math.h>
void add(int x,int y)
{
int results;
Result = x+Y;
print("Sum of %d and %d is %d.\n\n,x,y,result);
}
void main()
{

add(10,15);
add(500,325);
add(100,100005);
getch();
}
```

2. Based on the above, modify that program to operation of **subtract, multiply anddivide.**

20 (MARKS)

PROGRAMMING SOLUTION

ADDITION

(4 MARKS)

PRINT SCREEN RESULT COMMAND PROMPT

(1 MARK)

PROGRAMMING SOLUTION

SUBTRACT

(4 MARKS)

PRINT SCREEN RESULT COMMAND PROMPT

(1 MARK)

PROGRAMMING SOLUTION

MULTIPLY

(4 MARKS)

PRINT SCREEN RESULT COMMAND PROMPT

(1 MARK)

PROGRAMMING SOLUTION

DIVISION

(4 MARKS)

PRINT SCREEN RESULT COMMAND PROMPT

(1 MARK)

ANSWER THE QUESTIONS BELOW (5 MARKS)

Tick √ or X for the following questions.

1. In a function two return statement should never occur. ( )
2. A function may have any number of return statement each returning different values. ( )
3. A function cannot be defined inside another function. ( )
4. Function can be called either by value in reference. ( )
5. A Function can return floating point value. ( )

(5 MARKS)

DISCUSSION

.....(5 MARKS)

CONCLUSION

.....(5 MARKS)

REFERENCES

.....(5 MARKS)

NOTES

THANK YOU

REFERENCES

1. Syamsul Halim bin Wahab (2009). Asas Pengaturcaraan C Bagi Beginner. Jilid 1, (55-80).

2. Norizan Mohamad (2003). C++ Programming :Exercise Book. Jilid 1, (30-230).

3. Brian W.Kernighan Dennis M.Ritchie. The C Programming Language. 2nd Edition.(90-123).

4. C Programming Absolute Beginner's Guide. 3rd Edition. (78-232).

5. Noor Hasrina Bakar (2019). Programming in C For Foundation. (60-200).

6. Learn C Programming – Programiz. https://www.programiz.com › c-programming.

7. C Tutorial – Tutorialspoint. https://www.tutorialspoint.com › cprogramming

8. Learn C and C++ Programming. https://www.javatpoint.com

9. Learn C - Free Interactive C Tutorial.https://www.learn-c.org

e ISBN 978-967-2099-84-0

9 7 8 9 6 7 2 0 9 9 8 4 0