Laboratory Manual

For

Computer Programming in – C

(CT 213)

B.Tech (IT)

SEM II

June 2010

Faculty of Technology
Dharmsinh Desai University
Nadiad.
www.ddu.ac.in

Table of Contents

COMMON PROCEDURE

Tools / Apparatus: Unix/Linux/Windows Operating System, Text Editor, gcc compiler or Turbo C++ IDE

Procedure:

- Prepare the Box Diagram of the Program
- Prepare the Flow Chart of the Program
- Write the code of the program
- Compile the program for any compile-time errors
- Run the program
- Debug the program for any errors

Sample Experiment

1 AIM: Define a structure for items. The members are item number, item name, item price. Take all the details for at least 5 items. Using function search for the particular item by its name or by its number.

2 TOOLS/APPARATUS: Turbo C

3 STANDARD PROCEDURES:

COMMON PROCEDURE:

- Step 1: Create a folder in either E or F drive with your Id Number or Name Followed by RollNo.
- Step 2: Start the TC (Turbo C) from Desktop Icon or Go To Directory D:/TC/BIN/ and run tc.exe . An Editor will be start.
- Step 3: Click on File Menu --> New. New (.c) file will be created. Again Click on File -> Save an dialog box is going open write the path to your directory
  e.g. E:\structure\FileName.C and Press OK. Now your C program is going to save at your directory.
- Step 4: Go To Option->Directories Check That Include Directory is Set As D:\TC\Include and Library Directory is Set To D:\TC\LIB

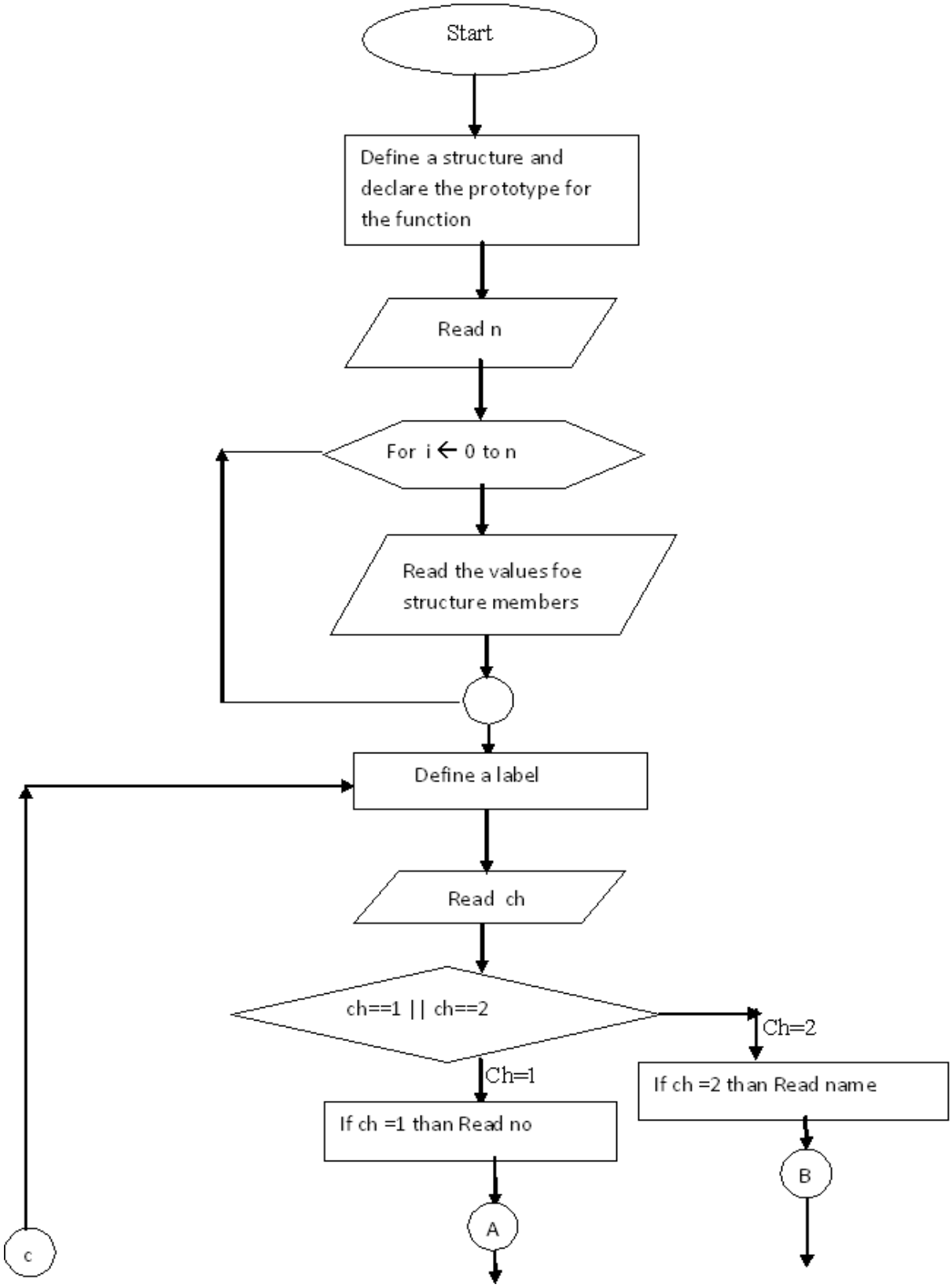
3.1 Analyzing the Problem:

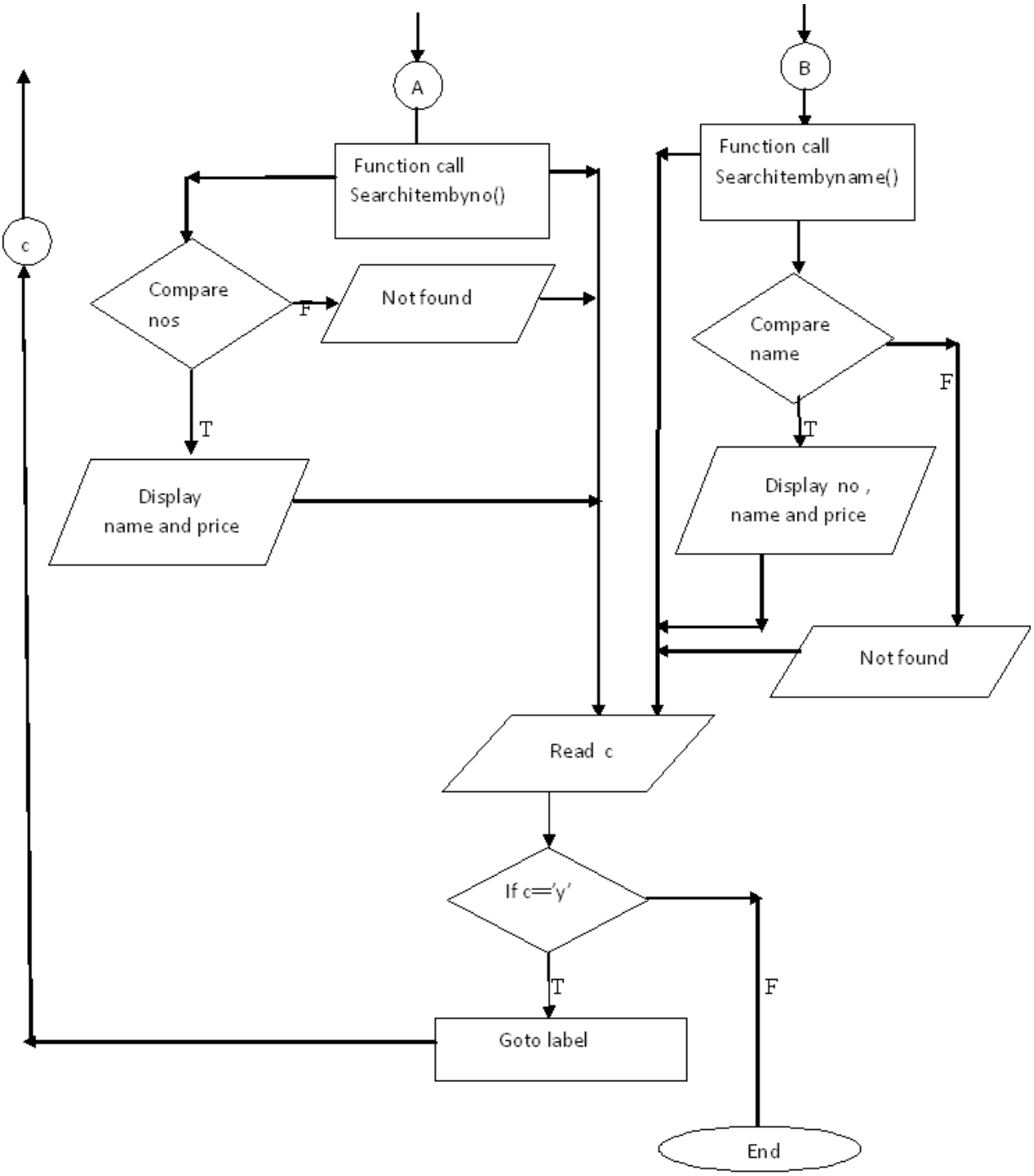
- First create a file named as “item.c” .
- After that includes the standard input/output files.
- Now define the structure “item” and its members.
- Create the functions which are necessary for the program.
- Now create the main function and take the information.

3.2 Designing the Solution:

1. Create a c file name as “item.c”.
2. Define the structure named as “item”. Also define its members that are item number, item name, item price.
3. Make a function to search a record by an item number that is “searchitembyno()” or by item name that is “searchitembyname()”.
4. Now in the main function declare a variable of the structure item “it[5]”. It should be an array because we want all the details for at least 5 items.
5. Now take the all the details by the user using scanf() function.
6. Now enter the choice by which you want to search. And call the function for desire output. And display the details using printf() function.

Flowchart:


Start
Define a structure and declare the prototype for the function
Read n
For i ← 0 to n
Read the values foe structure members
Define a label
Read ch
ch==1 || ch==2
Ch=2
If ch =2 than Read name
Ch=1
If ch =1 than Read no
B
A
c


A
B
c
Function call
Searchitembyno()
Function call
Searchitembyname()
Compare
nos
F
Not found
Compare
name
F
T
T
Display
name and price
Display no ,
name and price
Not found
Read c
If c='y'
T
F
Goto label
End

3.3 Implementing the Solution:
3.3.1 Writing Source Code:

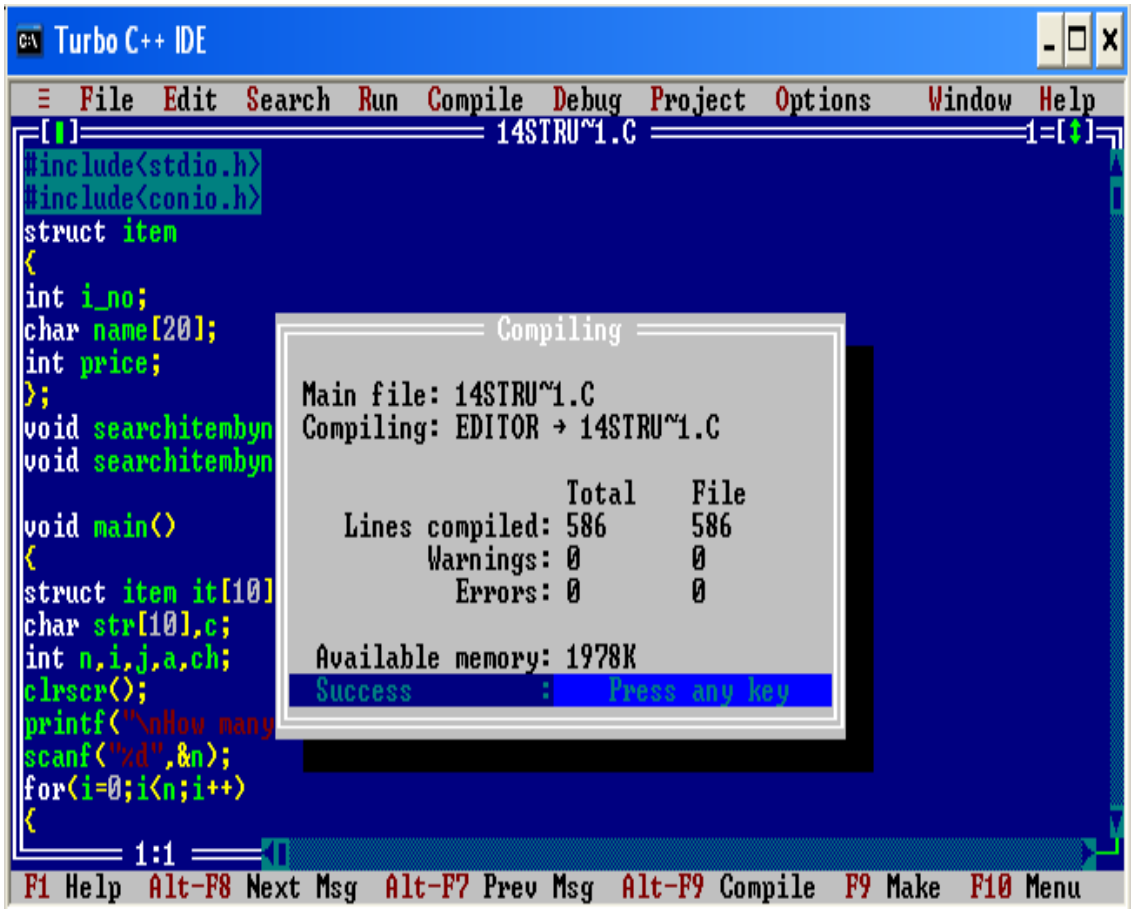
```
#include<stdio.h>
#include<conio.h>

struct item                  //defining the structure name item
{
int no;
char name[20];
int price;
};

void searchitembyno(struct item i1[ ] ,int,int);  //function prototype to search by item number

void searchitembyname(struct item i1[ ],char [ ],int); //function prototype to search by item name

void main( )
{
  struct item it[10],t;
  char str[10],c;
  int no,i,j,a,ch;
  clrscr();
  printf("\nHow many item :");
  scanf("%d",&n);              // take n number of items
  for(i=0;i<n;i++)
   {
      printf("\nEnter no, name and price of item :");
      scanf("\n%d %s %d",&it[i].i_no,it[i].name,&it[i].price);
 }

label:

// now for searching there are two options
printf("\n\nEnter 1 to search by number and 2 to search by name : ");
scanf("%d",&ch);

switch(ch)
{
case 1:   printf("\n\nEnter the no for the item u want to search : ");
          scanf("\n%d",&no);
          printf("\n");
          searchitembyno(it,a,no); //this is a function call to search by number
          break;
```

```
case 2:    printf("\n\nEnter the name of the item u want to search : ");
           scanf("\n%s",str);
           printf("\n");
           searchitembyname(it,str,n); //this is a function call to search by name
           break;
default:   break;
}

printf("\n want to search again ? y/n ");  //to search again Enter 'y' or to stop searching Enter 'n'.

c=getche();
if(c=='y')
{
 goto label ;
}

  getch();
}
//function definition ro search by number
void searchitembyno(struct item i1[],int a,int n)
 {
     int i,flag=0;
     for(i=0;i<n;i++)
     {
         if(i1[i].no==a)      //comparing nos
          {
            flag=1;
            printf("\n name and price is   %s  %d :",i1[i].name,i1[i].price);
            break;
          }
    }

   if(flag==0)
    {
      printf("\nitem not found ");
    }
}

//function definition ro search by name
void searchitembyname(struct item i1[],char s[],int n)
  {
     int i,flag=0;
     for(i=0;i<n;i++)
      {
          if((strcmp(i1[i].name,s))==0)    //comparing string
           {
```

```
            flag=1;
            printf("\n no name and price is %d  %s  %d :",i1[i].no,i1[i].name,i1[i].price);
            break;
          }
     }

    if(flag==0)
     {
       printf("\nitem not found ");
    }
}
```

3.3.2 Compilation /Running and Debugging the Solution:

- Go to Compile Menu and Press the Compile For the Compilation of the code.



- If Successful Compilation is done then Run the Code Using ctrl + F9 key.

```
How many item :5

Enter no, name and price of item :1
munch
10

Enter no, name and price of item :2
colgate
15

Enter no, name and price of item :3
pepsi
20

Enter no, name and price of item :4
rice
50

Enter no, name and price of item :5
notebook
15


Enter 1 to search by no and 2 to search by name : 1


Enter the no for the item u want to search : 4


 name and price is   rice  50 :
 want to search again ? y/n y

Enter 1 to search by no and 2 to search by name : 2


Enter the name of the item u want to search : pepsi


 no name and price is 3  pepsi  20 :
 want to search again ? y/n n
```

3.4 Testing the Solution:

- User must have entered all the details with respected to its data type.
- In search by name or number if that record is found than it will display the desire output. Otherwise it will display that item not found.
- If we enter a character instead of integer than the output will be unpredictable.
- Same way if we enter an integer instead of character than the output will be unpredictable.

4 Conclusions:

Hence I have conclude that this experiment will gives us the knowledge that how to code a meaningful and understandable program. As well as how to analyze, design, test and how to prepare the flowchart for the program.

Required Software/ Software Tool:
-Linux Operating System and/ or Windows Operating System
-Turbo C/C++ IDE.

Common procedure:

Step 1**:** For the given problem statement design
Flowchart/Algorithm/Logic
Step 2: Define variables and functions which will show the flow of program
Step 3: Write C code in the file with .c extension
Step 4: If the OS is Windows than compile and run .c file using turbo C.
Compile code using gcc compiler for Linux, which will create a.out executable file.
Step 5: Test the program using sample input and write down output.

TUTORIAL-1

- Answer the following questions.

1. Program & Programming Language
2. Compiler/Interpreter/Linker/Debugger
3. Algorithm/ Flowchart
4. Basic Structure of C Program.

EXPERIMENT-1

Aim: (A) Overview of Turbo C++ IDE.
(B) Overview of GCC.
(C) First Program: Display Message - "Hello World" on Standard Output.

Tools: Turbo C.

Procedure:

- Explain what the turbo C++ IDE is.
- Give the overview of GCC compiler.
- Make a C program to display the message "Hell World". Using printf function just displays the given message.

TUTORIAL-2

- Answer the following questions.

1. What is a comment? Why is it necessary?
2. Data types and use of *sizeof* operator
3. Classify briefly all operators
4. Difference between = and = =
5. What is the difference between prefix and postfix operators (i++, ++i)

EXPERIMENT-2

Aim: (A) Use of \n, \t and escape sequences.
(B) W.A.P to convert the temperature unit from Fahrenheit to Celsius using the formula C = (F – 32) / 1.8.
(C) Assume that any month is of 30 days. Now you are given total days. Find out the exact number of Years, Months & Days.
(D) You are given time in total seconds. Convert it into Hour: Min: Seconds format.

Tools: Turbo C.

Procedure:

- Explain the use of \n, \t and escape sequences.
- Make a C program in which converts the temperature taken by user in Fahrenheit to Celsius. And for conversion use the formula which is given.
- Make a program in which the total numbers of days are given and you have to convert that in perfect no of days, year and month.
- Same as the above program but here instead of days the seconds are given and you need to convert it into hour, min and seconds format.

TUTORIAL-3

- Answer the following questions.

1. Explain the conditional operator (exp1 ? exp2 : exp3).
2. Explain the different storage classes (auto, static, extern, register).
3. What is initialization? Why is it important?
4. What are enumeration variables?

EXPERIMENT-3

Aim: (A) W.A.P to determine whether input number is ODD or EVEN. Display appropriate message.
(B) W.A.P that will display Grade of student according to his/her marks using if else ladder. (Write same program using switch case also)

Marks > 80 then Grade = A
Marks between 61 & 80 then Grade = B
Marks between 51 & 60 then Grade = C
Marks between 41 & 50 then Grade = D
Marks between 35 & 40 then Grade = E
Marks < 35 then Grade = F

(C) W.A.P that computes and prints the Factorial of a given number.
(D) W.A.P that computes and prints the Fibonacci series.

Tools: Turbo C.

Procedure:

- Enter the number. And make the coding that if the entered number is odd than display the message the “No is odd”. Else display that “No is even”.
- Make a C code for grading a student based on the marks. You need to use the if else condition. And also make the same using switch case condition.
- Write a program to make the factorial of a given number. In that user enter a number and you have to find the factorial of a given number. (e.g. 5! =5 * 4 * 3 * 2 * 1)
- Write a program to generate the Fibonacci series for n terms. Where the value for n is given by the user.

TUTORIAL-4

- Answer the following questions.

1. Why the loop is required?
2. Explain Forward Jump.
3. Explain Backward Jump.
4. Explain Counter-Controlled and Sentinel-Controlled Loops.

EXPERIMENT-4

Aim: (A) W.A.P to print the following

```
*                    1
* *                  1 2
* * *                1 2 3
* * * *              1 2 3 4
* * * * *            1 2 3 4 5

 *                    1
 * *                  22
 * * *                333
 * * * *              4444
 * * * * *            55555
```

(B) W.A.P to count Blanks, Tabs and Newlines using *while and getchar.*
(C) Simple calculator using *do while.*
(D) W.A.P to check whether the input number is prime or not.
(E) W.A.P. to display even numbers between 2 to 20 without using the modulo operation.

Tools: Turbo C.

Procedure:

- Using loops generate all the patterns.
- Write a C code which will count the no of blanks, tabs and newlines in a program, using while and getchar. Here the string is taken by getchar function which is used to get a character from the user.
- Make a program in which no is entered by user and u need to check that no is prime or not. Divide the no by 2 and at the end if u get 1 than that no is prime else not.

  Write a C code which displays the even numbers between 2 to 20.

TUTORIAL-5

- Answer the following questions.

1. Explain the types of matrices:
    a. Single Dimensional
    b. Two Dimensional
    c. Multi Dimensional
2. Explain the different methods of declaring and defining Matrices.
3. Explain static arrays and dynamic arrays.

EXPERIMENT-5

Aim: (A) Find average of N numbers using arrays.
(B) Reading and displaying a 3X3 matrix.
(C) Addition of two 3X3 matrices.
(D) Multiplication of two matrices.

Tools: Turbo C.

Procedure:

- Create an array for n numbers where the value for n is given by user. Using 1-D array you can take the value and using loop you can fine the sum as well as average for n numbers.
- Using 2-D arrays and loops you can generate the 3x3 matrix.
- Create two 2-D arrays. After that using loop add the two matrices.
  Same way multiply the two 3x3 matrices based on multiplication rule in matrix.

TUTORIAL-6

- Answer the following questions.

1. Check validity for palindrome string.
2. Use of atoi and itoa functions.
3. Implement atoi and itoa logically in the main() itself.

EXPERIMENT-6

Aim: (A) W.A.P that will read a string and display its length as output (without using strlen( ) function).
(B) Write a program that will read two strings.
If given two strings are same then print "Given strings are same".
If given two strings are not same then print "Given strings are not same"
(C) Write a program that will concatenate one string to the end of another string and display it as output.

Tools: Turbo C.

Procedure:

- Using loop take the string from the user. And just increment the value of the counter by 1 for each and every character. And find the length of the string.
- Take two strings from the user using loop. After that compare each and every character of one string with another string. If the strings are same than display the respective message.
- Take two strings from the user. Now at the end of the 1st string you have to attach another string.

TUTORIAL-7

- Answer the following questions.

1. Printline() function with two arguments: the character to be displayed and Number of times.
2. Write a function IsPrime() which returns true if the number is prime. otherwise false(use of enum Boolean is preferred)

EXPERIMENT-7

Aim: (A) Printline() which prints '=' sign 81 times in the same line.
(B) Write a function to calculate and display the total amount given that

Total amount = $p * (1 + r)^2$

Where p = principle amount

r= rate of interest

n = period.

(C) Modify above program for returning total amount.

Tools: Turbo C.

Procedure:

- Make a function which prints the '=' sign 81 times.
- Make functions which calculate the total amount by the formula which is given.
- Modify the above program which returns the total amount.

TUTORIAL-8

- Answer the following questions.

1. Find minimum and maximum from list of integers. Use function and array as
2. arguments.
3. Write a program to generate and display n terms of fibbonaci series using recurcive function.

EXPERIMENT-8

Aim: (A) Factorial of n using recursive function.
(B) Sorting of numbers in ascending order using function with array as arguments, using the following methods:

- Bubble Sort
- Selection Sort
- Insertion Sort

Tools: Turbo C.

Procedure:

- Make a recursive function for factorial.
- Make a program which sort the given numbers using function in which array is given as an argument.

TUTORIAL-9

- Answer the following questions.

1. Write a function orig_str_rev() which reverses the original string passed to it as arguments. Use pointers.
2. Implement mystrncpy() which copies the first n characters of source string to destination string. Use pointers.
3. Sorting of strings using pointers. Do not sort actual strings from their memory. Just manipulate pointers.

EXPERIMENT-9

Aim: (A) Swap/exchange values of two integer variables using function. Use pointer.
(B) Write a function mystrcat(s,t) which copies string t to the end of the string s. Use pointers.
(C) Write a stringindex() function which takes two arguments. One argument is the subject string and second is pattern string. The function searches for the pattern in the subject string and returns the position/index of the start of the string where match found. The function returns -1 if the pattern is not found. Use pointers.

Tools: Turbo C.

Procedure:

- Swap the values of the two integers using function. Give the integers as an argument to the function.
- Same way make a function for concatenation of two strings.
- Make a stringindex function which finds for the pattern string in the original string and returns the position for the match.

TUTORIAL-10

- Answer the following questions.

1. Represent a point p(x,y) using structure. Write a function addpoint(..) to add two points and return the result point.
2. Same way write another function ptinrect() with two arguments a point and a rectangle , it returns true if the point within the rectangle otherwise returns false.

EXPERIMENT-10

Aim: (A) you are supposed to generate a result table which consists of student id, student name, marks of three subject and total marks. Write a program which takes input for ten students and displays result table. Also display student information separately who got the highest total. USE STUCTURES.
(B) We suppose to store information of 10 persons. Information includes name and age. But criteria is for the child age should be in form of full birth date, for an adult the age should be in years only, while for aged person store age indicating the status 'O'. Use union for memory efficiency.

Tools: Turbo C.

Procedure:

- Make a structure for student details. Write a program in which take the input for ten students and display the result table for each and every student.
- Store the information for 10 people. And display the status of age.

TUTORIAL-11

- Answer the following questions.

1. Write a program which calculates no. of characters, words, and lines of the input file. Use command line argument to get the input file name;
2. Separate integer file into two different Odd.txt and Even.txt which contains the corresponding odd or even numbers.

EXPERIMENT-11

Aim: (A) Read input characters from user and write it to file. After that read the content of the same file and display on screen.
(B) Implement copy command with the use of command line arguments. Like mycopy source.txt destination.txt copies content of source file to destination file.
(C) Write integer numbers to a file. Read them back and find average of all numbers.

Tools: Turbo C.

Procedure:

- Take the input from the user and using functions available for the file write the data to the file. After that again read the data from the file and displays it on the screen.
- Use the command line argument to read the argument given by the user and using that make the program which copy the content of one file to the another file.
- Make a program which takes the input from the user for integer values write this values to a file. Again read it back and find the average.

TUTORIAL-12

- Answer the following questions.

1. Describe different types of linked lists.
2. What are the differences between arrays and linked lists?

EXPERIMENT-12

Aim: (A) WAP to implement Stack using Linked Lists.
(B) WAP to implement Queue using Linked Lists.

Tools: Turbo C.

Procedure:

- Write a code which implement stack. You need to use linked list.
- Write a code which implement queue. You need to use linked list.

TUTORIAL-13

- Answer the following questions.

1. What are the advantages of using linked lists?

EXPERIMENT-13

Aim: (A) WAP to implement Single Linked List.
(B) WAP to implement Doubly-Linked List.

Tools: Turbo C.

Procedure:

- Write a code which implement single linked list. In which one node is point to the next node. Create a node using structure.
- Write a code which implement doubly linked list. Here the node will point both the side. Create the node using structure.

References

Reference Books

- Programming in ANSI C by Balagurusamy, 2nd Ed., TMH
- http://www.oopweb.com/CPP/Download/crc.zip
- http://www.freebookcentre.net/programming-books-download/C-Programming-Tutorial.html