1. **Contiguous allocation of file**

#include<stdio.h>

#include<conio.h>

void main()

{

int f[50], i, st, len, j, c, k, count = 0;

for(i=0;i<50;i++)

f[i]=0;

printf("Files Allocated are : \n");

x:

count=0;

printf("Enter starting block and length of files: ");

scanf("%d%d", &st,&len);

for(k=st;k<(st+len);k++)

if(f[k]==0)

count++;

if(len==count)

{

for(j=st;j<(st+len);j++)

if(f[j]==0)

{

f[j]=1;

printf("%d\t%d\n",j,f[j]);

}

if(j!=(st+len-1))

printf("The file is allocated to disk\n");

}

else

printf("The file is not allocated \n");

printf("Do you want to enter more file(Yes - 1/No - 0)");

scanf("%d", &c);

if(c==1)

goto x;

else

exit(0);

}

1. **Write a c program to create directory and file in any drive**

#include <windows.h>

#include <stdio.h>

#include <stdlib.h>

#define MAX\_NAME\_LENGTH 260

void create\_directory(char path[]) {

BOOL success = CreateDirectory(path, NULL);

if (!success) {

printf("Failed to create directory: %s\n", path);

exit(1);

}

}

void create\_file(char path[]) {

HANDLE file\_handle = CreateFile(path, GENERIC\_READ | GENERIC\_WRITE, 0, NULL, CREATE\_NEW, FILE\_ATTRIBUTE\_NORMAL, NULL);

if (file\_handle == INVALID\_HANDLE\_VALUE) {

printf("Failed to create file: %s\n", path);

exit(1);

}

CloseHandle(file\_handle);

}

void create(char path[], int is\_directory) {

if (is\_directory) {

create\_directory(path);

} else {

create\_file(path);

}

}

int main() {

create("C:\\test\_directory", 1);

create("C:\\test\_directory\\test\_file.txt", 0);

return 0;

}

OR

**Write a program to create directory and file**

#include <stdio.h>

#include <stdlib.h>

#include <sys/stat.h>

#include <sys/types.h>

#include<unistd.h>

#include <fcntl.h>

int main() {

char dirname[] = "new\_directory";

char filename[] = "new\_file.txt";

int dir\_result, file\_result;

// Create a new directory

dir\_result = mkdir(dirname, S\_IRWXU);

if (dir\_result == -1) {

printf("Failed to create directory!\n");

exit(EXIT\_FAILURE);

}e

printf("Directory created successfully!\n");

// Create a new file in the new directory

chdir(dirname);

file\_result = creat(filename, S\_IRWXU);

if (file\_result == -1) {

printf("Failed to create file!\n");

exit(EXIT\_FAILURE);

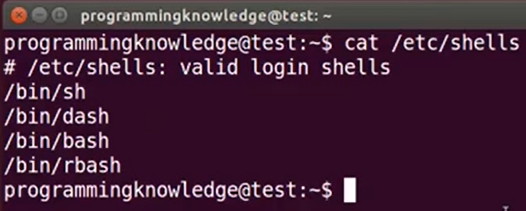
}

printf("File created successfully!\n");

return 0;

}

1. **Bash Shells programming :** 
   1. Open terminal
   2. Cat /etc/shells

You can see different type of shells 

Where ,

Sh is born shells , original shells

Bash born again shells , reinvented from born shells

Standard bash shells

* 1. To see the location of bash
     1. Which bash
  2. Create new file
     1. Touch hello.sh

It open file

* + 1. Ls –al

List permission assign to file

#! /bin/bash

Echo “hello world”

Save it and execute it

./hello.sh

To give permission

Chmod +X hello.sh