24. Write a C Program to find the maximum size of a file that can be stored in the below file system that uses inodes to represent files. Disk blocks are 8 KB in size, and a pointer to a disk block requires 4 bytes. This file system has 12 direct disk blocks, as well as single, double, and triple indirect disk blocks.

Test Case:

● Check that the start blocks and the required contiguous blocks are free.

● If free, allocate those blocks to the file.

● If not free, find the next available contiguous blocks.

Program:-

#include <stdio.h>

#include <stdlib.h>

#define BLOCK\_SIZE 8192 // 8 KB disk block size

#define POINTER\_SIZE 4 // Size of a pointer to a disk block

#define DIRECT\_BLOCKS 12 // Number of direct disk blocks

#define POINTERS\_PER\_BLOCK (BLOCK\_SIZE / POINTER\_SIZE)

// Function to calculate the maximum file size

size\_t calculateMaxFileSize() {

size\_t maxFileSize = 0;

// Calculate the maximum size of direct blocks

maxFileSize += DIRECT\_BLOCKS \* BLOCK\_SIZE;

// Calculate the maximum size of single indirect blocks

size\_t singleIndirectBlocks = POINTERS\_PER\_BLOCK;

maxFileSize += singleIndirectBlocks \* BLOCK\_SIZE;

// Calculate the maximum size of double indirect blocks

size\_t doubleIndirectBlocks = POINTERS\_PER\_BLOCK \* POINTERS\_PER\_BLOCK;

maxFileSize += doubleIndirectBlocks \* BLOCK\_SIZE;

// Calculate the maximum size of triple indirect blocks

size\_t tripleIndirectBlocks = POINTERS\_PER\_BLOCK \* POINTERS\_PER\_BLOCK \* POINTERS\_PER\_BLOCK;

maxFileSize += tripleIndirectBlocks \* BLOCK\_SIZE;

return maxFileSize;

}

int main() {

size\_t maxFileSize = calculateMaxFileSize();

printf("Maximum file size in the given file system: %lu bytes\n", maxFileSize);

}

Output:-

