24. DISK BLOCKS

#include <stdio.h>

#include <stdlib.h>

#define BLOCK\_SIZE 8192

#define POINTER\_SIZE 4

#define DIRECT\_BLOCKS 12

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

#define SINGLE\_INDIRECT\_POINTERS POINTERS\_PER\_BLOCK

#define DOUBLE\_INDIRECT\_POINTERS (POINTERS\_PER\_BLOCK \* POINTERS\_PER\_BLOCK)

#define TRIPLE\_INDIRECT\_POINTERS (POINTERS\_PER\_BLOCK \* POINTERS\_PER\_BLOCK \* POINTERS\_PER\_BLOCK)

long long calculateMaxFileSize()

{

long long maxFileSize = 0;

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

long long singleIndirectBlocks = POINTERS\_PER\_BLOCK;

maxFileSize += singleIndirectBlocks \* BLOCK\_SIZE;

long long doubleIndirectBlocks = POINTERS\_PER\_BLOCK \* POINTERS\_PER\_BLOCK;

maxFileSize += doubleIndirectBlocks \* BLOCK\_SIZE;

long long tripleIndirectBlocks = POINTERS\_PER\_BLOCK \* POINTERS\_PER\_BLOCK \* POINTERS\_PER\_BLOCK;

maxFileSize += tripleIndirectBlocks \* BLOCK\_SIZE;

long long singleIndirectPointers = singleIndirectBlocks \* POINTERS\_PER\_BLOCK;

long long doubleIndirectPointers = doubleIndirectBlocks \* POINTERS\_PER\_BLOCK + singleIndirectPointers;

long long tripleIndirectPointers = tripleIndirectBlocks \* POINTERS\_PER\_BLOCK + doubleIndirectPointers;

maxFileSize += singleIndirectPointers \* BLOCK\_SIZE;

maxFileSize += doubleIndirectPointers \* BLOCK\_SIZE;

maxFileSize += tripleIndirectPointers \* BLOCK\_SIZE;

return maxFileSize;

}

int main()

{

long long maxFileSize = calculateMaxFileSize();

printf("Maximum size of a file in the file system: %lld bytes\n", maxFileSize);

return 0;

}

OUTPUT

