Code (Available at the end as text) : 

a) yes, it contains new line, as indicated by



b) SPI’s are always at the beginning, as indicated by 

c) It writes to the end. Append writes to the end of the file, regardless of SPI’s location.

d) Yes. Similar to touch in unix systems, a file can be empty but still remain in existence.

Code:

#include <stdio.h>

#include "fileTest.h"

void initFile();

void spaceCheck();

void SPICheck();

void spaceCheck();

void moveSPIandAppend();

int main(int argc, char \*argv[]){

initFile(); //create file and write

spaceCheck(); //check if newline exists

SPICheck(); //check for SPI position

moveSPIandAppend(); //move SPI and append

}

void initFile(){

char toAdd[] = "Hello\n";

FILE \*fptr = fopen("createdFile.txt","w");

if(fptr==NULL){

printf("Could not open file. \n");

}else{

fprintf(fptr,toAdd);

fclose(fptr);

}

}

void SPICheck(){

FILE \*fptr = fopen("createdFile.txt","r");

if(fptr==NULL){

printf("Could not open file. \n");

}else{

long fSize = ftell(fptr);

printf("SPI is currently at offset %ld \n",fSize);

}

fclose(fptr);

}

void spaceCheck(){

FILE \*fptr = fopen("createdFile.txt","r");

char buff[BUFSIZ];

char cur = fgetc(fptr);

while(cur!=EOF&&cur!='\n'){

if(cur!='\n'){

cur=fgetc(fptr);

}

}

if(cur=='\n'){

printf("Yes, file contains newline.\n");

}else{

printf("No, file does not contain newline. \n");

}

fclose(fptr);

}

void moveSPIandAppend(){

FILE \*fptr = fopen("createdFile.txt","a");

fseek(fptr,0L,SEEK\_SET);

char toWrite[] = "Goodbye CSCD240\n";

fprintf(fptr,toWrite);

fclose(fptr);

}