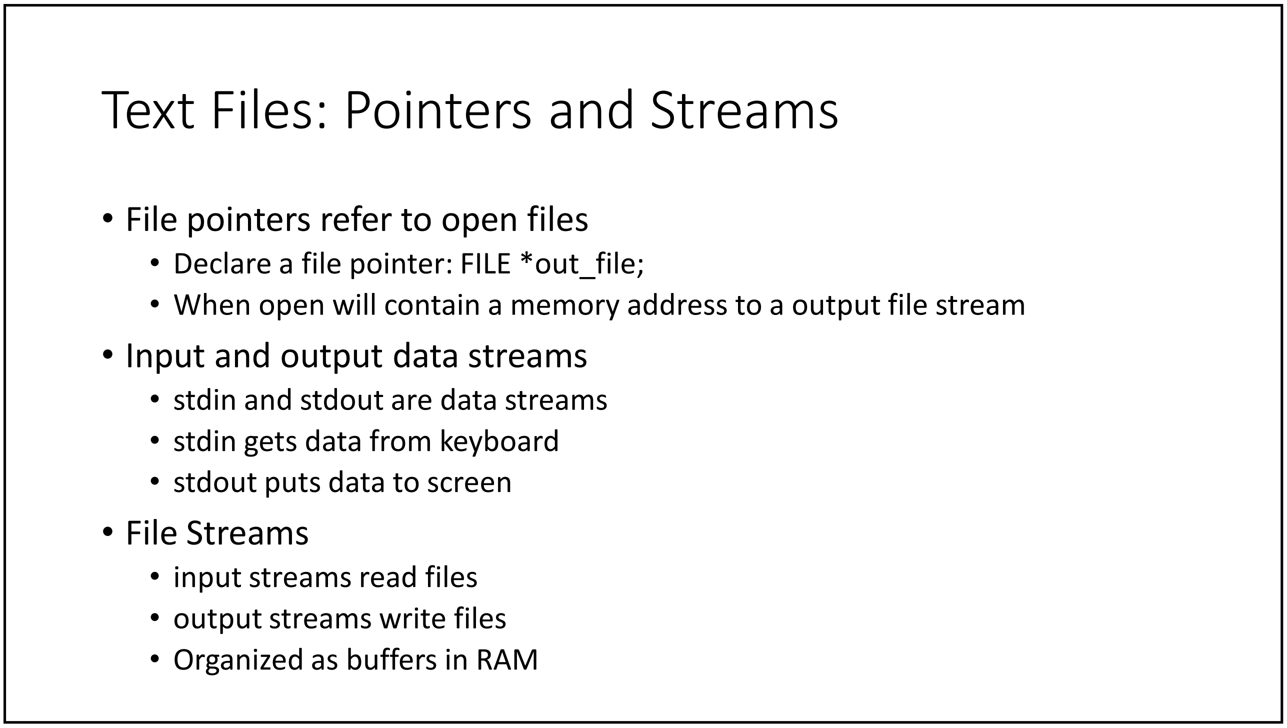
Text and Binary File Pointers

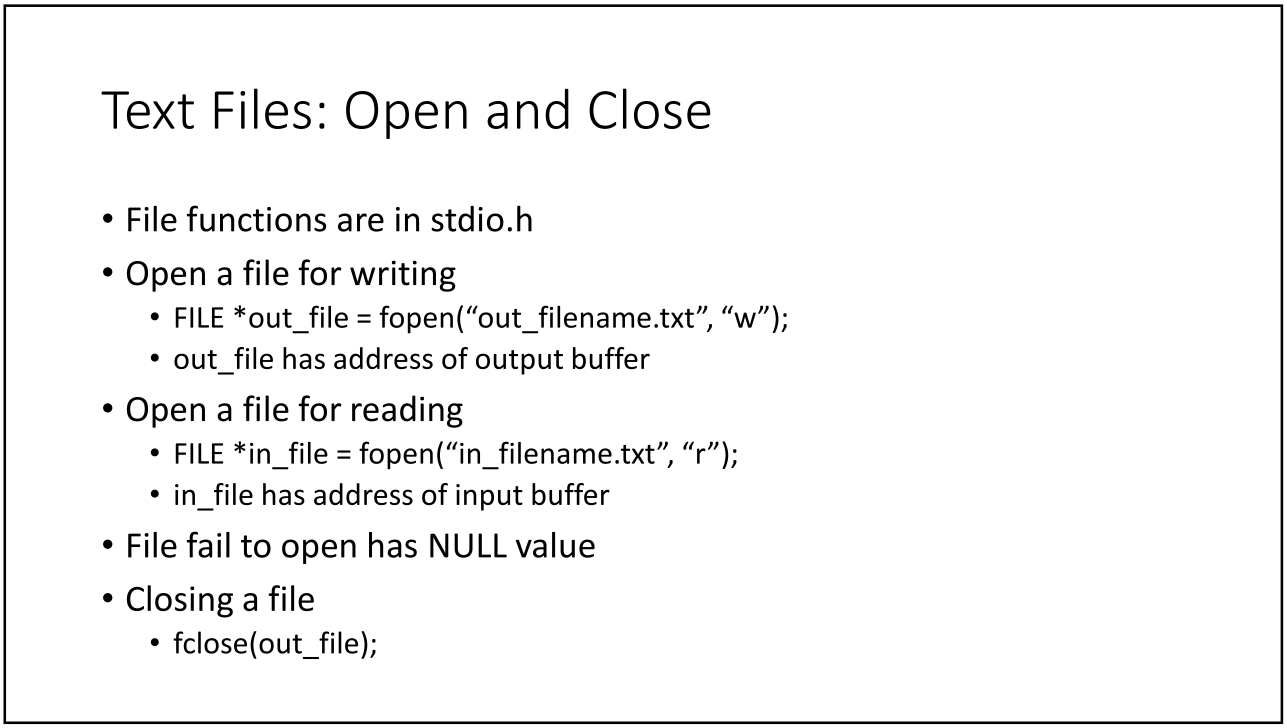
Text Files: Contents and Formatting

- Text files contain characters
  - Lines contain characters
  - Lines end with ‘\n’
  - File ends with end-of-file market (EOF)
- Formatting files for write operations is similar to printf
  - Same placeholders (%d, %c, %f, etc.)
  - Same escape sequences (/n, /t, //, etc.)



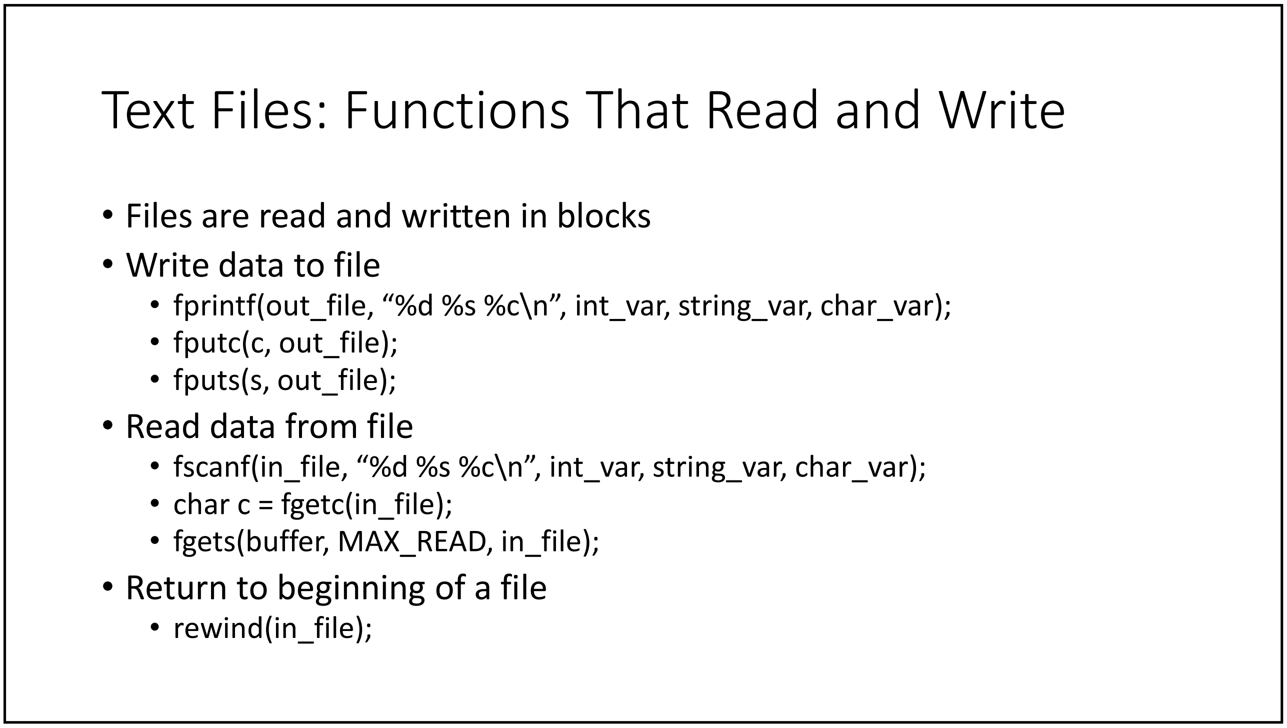
Text Files: Pointers and Streams

- File pointers refer to open files
  - Declare a file pointer: FILE *out_file;
  - When open will contain a memory address to a output file stream
- Input and output data streams
  - stdin and stdout are data streams
  - stdin gets data from keyboard
  - stdout puts data to screen
- File Streams
  - input streams read files
  - output streams write files
  - Organized as buffers in RAM



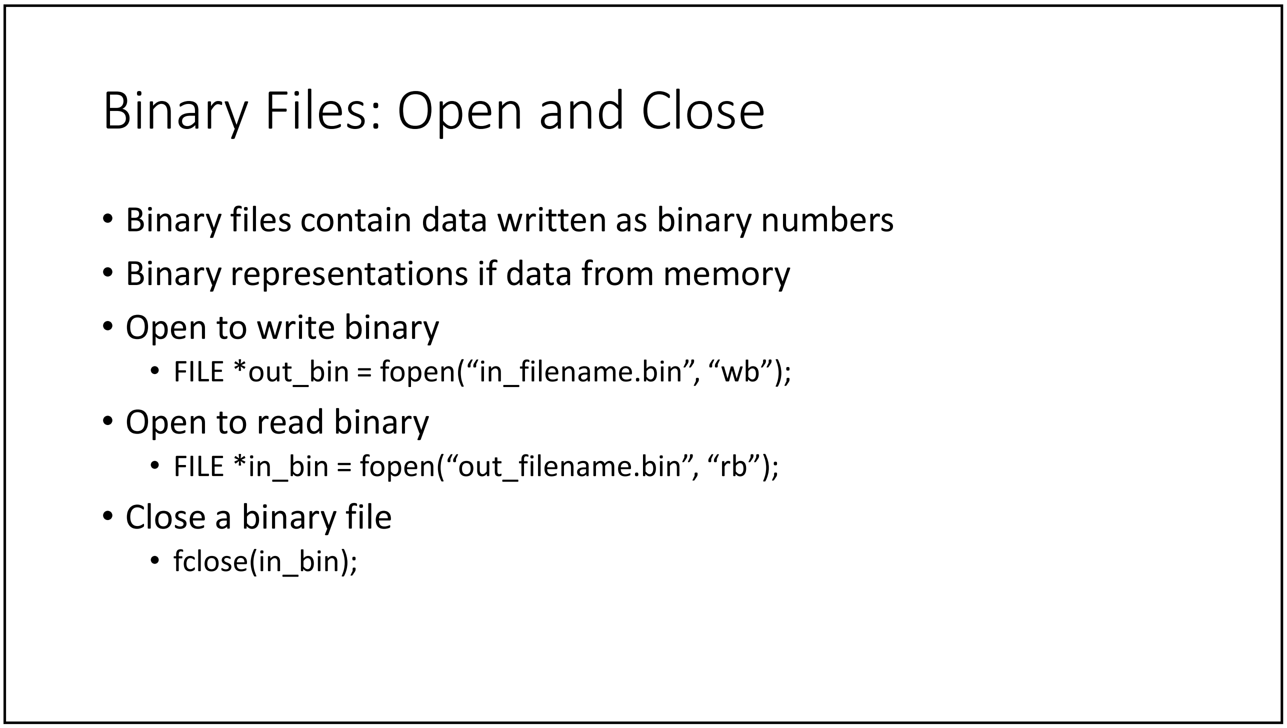
Text Files: Open and Close

- File functions are in stdio.h
- Open a file for writing
  - FILE *out_file = fopen(“out_filename.txt”, “w”);
  - out_file has address of output buffer
- Open a file for reading
  - FILE *in_file = fopen(“in_filename.txt”, “r”);
  - in_file has address of input buffer
- File fail to open has NULL value
- Closing a file
  - fclose(out_file);



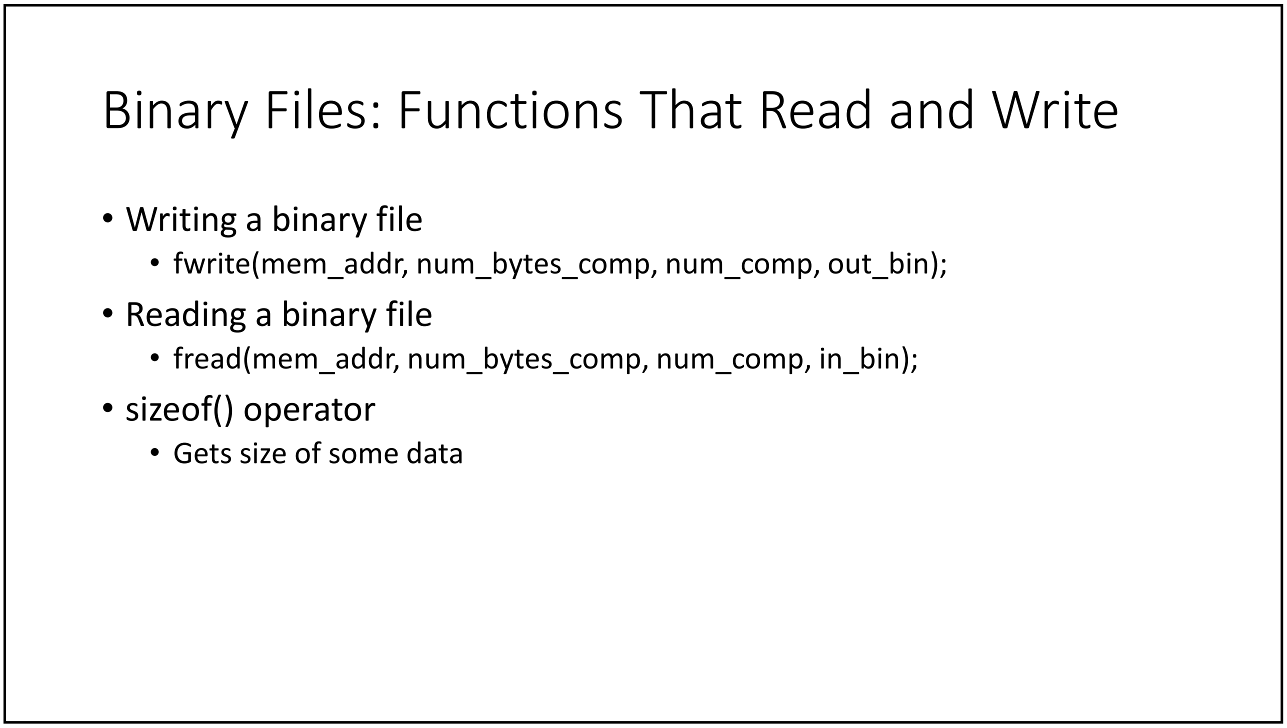
Text Files: Functions That Read and Write

- Files are read and written in blocks
- Write data to file
  - fprintf(out_file, “%d %s %c\n”, int_var, string_var, char_var);
  - fputc(c, out_file);
  - fputs(s, out_file);
- Read data from file
  - fscanf(in_file, “%d %s %c\n”, int_var, string_var, char_var);
  - char c = fgetc(in_file);
  - fgets(buffer, MAX_READ, in_file);
- Return to beginning of a file
  - rewind(in_file);



Binary Files: Open and Close

- Binary files contain data written as binary numbers
- Binary representations if data from memory
- Open to write binary
  - FILE *out_bin = fopen(“in_filename.bin”, “wb”);
- Open to read binary
  - FILE *in_bin = fopen(“out_filename.bin”, “rb”);
- Close a binary file
  - fclose(in_bin);



Binary Files: Functions That Read and Write

- Writing a binary file
  - fwrite(mem_addr, num_bytes_comp, num_comp, out_bin);
- Reading a binary file
  - fread(mem_addr, num_bytes_comp, num_comp, in_bin);
- sizeof() operator
  - Gets size of some data