Etude #13 - Bug Squashing

Problems with makefile

- Don't throw errors to null, instead errors can just be printed to stdout
- -lm is unnecessary
- I would have also added -O2 for recommended optimisations, but I've assumed that this program is still being developed on.
- I did add -Wall for recommended compiler warnings though.

Problems with main.c

Basic Syntax Errors

- #include <stdio.h> is included twice. I've taken the second line out.
- Email address is spelt wrong in struct S.
- The asterisk should be directly to the left of the char variables – firstName, lastName and emailAddress for pointers.
- The search functions (changed names of functions to findbyfn, findbyln, findbyem, findbyph) have been changed so that the variable I is set to zero when it is called and incremented each iteration through the while loop. Because it is two strings (or character arrays more specifically) being compared, a simple greater than or less than operator will do nothing. The strcmp() function, which is built into the <string.h> module is used for this purpose.
- The file opening was moved to the top of the main method because C forbids mixing declarations and code, same with – int command = 10;

Semantics/Readability

- All of the methods are ambiguously named.
  - I've named them according to what it looks like they're doing.
- The struct 'S' is ambiguously named, it seems to represent a person, so I've changed it to person.
- In the switch statement, last name is spelt wrong.
- In the switch statement, there are two 'looking for email?' printf statements. The second one can be changed to 'looking for phone number?'.
- Printing out zero and one in response to searches is not very user friendly, print out yes or no instead using a function that takes an input and returns a string called "yes" or a string called "no".

Error Checking, User-Friendliness

- Main method does not return anything at the end.
- In the fscanf statement, the strings are referenced with an ampersand, which is not required because strings are character arrays, unlike integers which are not pointers at all. Because phone number could have special digits or be very long, I've changed phone number from an integer to a string.
- If there are no arguments provided, I print a message to stderr and exit the program.
- How is the user supposed to know what the commands do? I print out instructions so the user knows what commands do what.

Input/Output

- What happens if the file is not there? The fopen command returns a pointer to a file, if the file is not there, the function will return NULL. Because of this, I check if the file pointer points to NULL and if so, I print a message to stderr using the fprintf() function as well as exiting the program.
- Fscanf is not secure so I use getline() for getting a line of input from the file input. I get the line of text, put it in the buffer variable, and use strtok() to break the line up into first name, last name, email, and phone number. This data is then entered into the people struct.
- The file has to be closed with fclose().

Memory

- Most of the memory that was allocated early in the program is not freed, so at the bottom of the main function, I free the individuals persons and their properties as well as the overall array of structs – people.
- After running I got a segmentation fault, even without input. A segmentation fault means you are trying to access memory not allocated to your program. This was occurring because memory wasn't being allocated properly.