```c
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

typedef struct User {
  char* username;
  char passwd_sha[32];
} User;

static User users[1000];

void load_users(char* path) {
  FILE* f = fopen(path, "r");
  char buffer[10000];
  int i = 0;
  while(fgets(buffer, 10000, f) != NULL) {
    char* username_part = strtok(buffer, " ");
    char* password_part = strtok(NULL, " ");

    // buffer is reused, so need to make some
    // space for the username for use in the struct
    char* username = malloc(strlen(buffer));
    strcpy(username, username_part);
    printf("%s@%p\n", username, username);

    // Make a User struct with that username,
    // and then copy the (fixed-length) hash part into it
    User current_user = { username, {} };
    strncpy(current_user.passwd_sha, password_part, 32);

    users[i] = current_user;
    i += 1;

    // done with username, so free it now ⚠️
    free(username);
  }
}

int main() {
  load_users("users.txt");
  char* username = malloc(7);
  printf("Enter your username: ");
  fgets(username, 6, stdin);
  username[strcspn(username, "\n")] = '\0';
  for(int i = 0; i < 1000; i += 1) {
    char* username = users[i].username;
    if(username == NULL) { break; }
    printf("%s@%p: %.32s\n", username, username, users[i].passwd_sha);
  }
}
```

```
> ./login
jpolitz@0x102e31b10
gsoosairaj@0x102e31b10
aschulman@0x102e31b10
Enter your username: bob
bob@0x102e31b10: abcdef1234567890abcdef1234567890
bob@0x102e31b10: 1234567890abcdef1234567890abcdef
bob@0x102e31b10: 9876543210abcdef9876543210abcdef
```

```
jpolitz abcdef1234567890abcdef1234567890
gsoosairaj 1234567890abcdef1234567890abcdef
aschulman 9876543210abcdef9876543210abcdef
```

users.txt