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

#define MAX_ENTRY_LENGTH 1000

// Function to add an entry
void add_entry() {
   FILE *file = fopen("diary.txt", "a");
   if (!file) {
       printf("Error opening file!\n");
       return;
   }

   char entry[MAX_ENTRY_LENGTH];
   getchar();

   printf("Enter your diary entry:\n");
   fgets(entry, MAX_ENTRY_LENGTH, stdin);

   fprintf(file, "------------------------------\n");
   fprintf(file, "%s", entry);
   fprintf(file, "------------------------------\n\n");

   fclose(file);
printf("Entry added successfully!\n");
}

// Function to view entries
void view_entries() {
   FILE *file = fopen("diary.txt", "r");
   if (!file) {
       printf("No diary entries found.\n");
       return;
   }

   char ch;

   printf("\n=== ALL DIARY ENTRIES ===\n\n");
   while ((ch = fgetc(file)) != EOF) {
       putchar(ch);
   }

   fclose(file);
}
// Function to search entries by keyword
void search_entry() {
   FILE *file = fopen("diary.txt", "r");
   if (!file) {
       printf("No diary entries found!\n");
       return;
   }

   char keyword[100];
   char line[1000];
   int found = 0;

   getchar();
   printf("Enter keyword to search: ");
   fgets(keyword, sizeof(keyword), stdin);
   keyword[strcspn(keyword, "\n")] = 0;

printf("\n=== SEARCH RESULTS ===\n\n");

while (fgets(line, sizeof(line), file)) {
       if (strstr(line, keyword)) {
           found = 1;
           printf("%s", line);
       }
   }

   if (!found) {
       printf("No matching records found.\n");
   }

   fclose(file);
}

// Function to delete an entry by keyword
void delete_entry() {
   FILE *file = fopen("diary.txt", "r");
   if (!file) {
       printf("No entries available to delete!\n");
       return;
   }
FILE *temp = fopen("temp.txt", "w");


if (!temp) {
       printf("Error creating temp file!\n");
       return;
   }

   char keyword[100];
   char line[1000];
   int skip_block = 0, deleted = 0;

   getchar();
   printf("Enter keyword to delete entry: ");
   fgets(keyword, sizeof(keyword), stdin);
   keyword[strcspn(keyword, "\n")] = 0;

   while (fgets(line, sizeof(line), file)) {
       if (strstr(line, keyword)) {
           skip_block = 1;
           deleted = 1;
       }

       if (skip_block && strstr(line, "------------------------------")) {
           skip_block = 0;
           continue;
       }
if (!skip_block)
           fputs(line, temp);
   }

   fclose(file);
   fclose(temp);

   remove("diary.txt");
   rename("temp.txt", "diary.txt");

   if (deleted)
       printf("Entry deleted successfully.\n");
   else
       printf("No matching entry found.\n");
}

// Main Menu
void display_menu() {
   printf("\n===== Department Diary =====\n");
   printf("1. Add Entry\n");
   printf("2. View Entries\n");
   printf("3. Search Entry\n");
   printf("4. Delete Entry\n");
   printf("5. Exit\n");
   printf("Choose an option: ");
}
int main() {
   int choice;

   while (1) {
       display_menu();
       scanf("%d", &choice);

       switch (choice) {
           case 1:
               add_entry();
               break;
           case 2:
               view_entries();
               break;
           case 3:
               search_entry();
               break;
           case 4:
               delete_entry();
               break;
           case 5:
               printf("Exiting program.\n");
```

```
                exit(0);
            default:
                printf("Invalid option! Try again.\n");
        }
    }

    return 0;
}
```