1b…#include <string.h>

```
// Structure to represent a day
struct Day {
    char *name;      // dynamically allocated string for day name
    int date;        // integer date
    char *activity;  // dynamically allocated string for activity
};

// Function prototypes
struct Day* create(int size);
void read(struct Day *calendar, int size);
void display(struct Day *calendar, int size);

int main() {
    int size = 7;
    struct Day *calendar;

    // Step 1: Create the calendar (dynamic memory allocation)
    calendar = create(size);

    // Step 2: Read data for 7 days
    read(calendar, size);

    // Step 3: Display the weekly report
    display(calendar, size);

    // Free the allocated memory
    for (int i = 0; i < size; i++) {
        free(calendar[i].name);
        free(calendar[i].activity);
    }
    free(calendar);

    return 0;
}

// Function to create dynamic array for 7 days
struct Day* create(int size) {
    struct Day *calendar;
    calendar = (struct Day *)malloc(size * sizeof(struct Day));
    if (calendar == NULL) {
        printf("Memory allocation failed!\n");
        exit(1);
```

```
    }
    return calendar;
}

// Function to read details of each day
void read(struct Day *calendar, int size) {
    char tempName[50], tempActivity[100];

    printf("Enter details for %d days:\n", size);

    for (int i = 0; i < size; i++) {
        printf("\nDay %d:\n", i + 1);

        printf("Enter name of the day: ");
        scanf(" %[^\n]", tempName);

        printf("Enter date (e.g., 1-31): ");
        scanf("%d", &calendar[i].date);

        printf("Enter activity description: ");
        scanf(" %[^\n]", tempActivity);

        // Allocate and copy strings
        calendar[i].name = (char *)malloc(strlen(tempName) + 1);
        strcpy(calendar[i].name, tempName);

        calendar[i].activity = (char *)malloc(strlen(tempActivity) + 1);
        strcpy(calendar[i].activity, tempActivity);
    }
}

// Function to display the week's calendar report
void display(struct Day *calendar, int size) {
    printf("\n===== Weekly Calendar Report =====\n");
    for (int i = 0; i < size; i++) {
        printf("\nDay: %s\n", calendar[i].name);
        printf("Date: %d\n", calendar[i].date);
        printf("Activity: %s\n", calendar[i].activity);
    }
}
```