```c
#include <stdio.h>

struct student{
    int r_no;
    char name[20];
    char course[20];
    int fees;
};

int main() {
    struct student stud1,*ptr;
    ptr=&stud1;
    printf("Enter the details of the student:\nRoll Number: ");
    scanf("%d",&ptr->r_no);
    printf("Name: ");
    scanf("%s",ptr->name);
    printf("Course: ");
    scanf("%s",ptr->course);
    printf("Fee: ");
    scanf("%d",&ptr->fees);
    printf("\n\nDETAILS OF THE STUDENT\nROLL NUMBER: %d\n",ptr->r_no);
    printf("NAME: %s\n",ptr->name);
    printf("COURSE: %s\n",ptr->course);
    printf("FEES: %d\n",ptr->fees);
    return 0;
```

A function declaration without a p

```
Enter the details of the student:
Roll Number: 221011
Name: Siti
Course: Programming
Fee: 46.50


DETAILS OF THE STUDENT
ROLL NUMBER: 221011
NAME: Siti
COURSE: Programming
FEES: 46
Program ended with exit code: 0
```

```c
#include <stdio.h>

typedef struct distance {
    int kms;
    int meters;
} DISTANCE;

DISTANCE addDistance(DISTANCE d1, DISTANCE d2);

int main() {
    DISTANCE d1 = {0, 0}, d2 = {0, 0}, d3 = {0, 0};
    int option;
    int inputFlag = 0;

    do {
        // Display the main menu
        printf("********MAIN MENU**********\n");
        printf("1. Input distances\n");
        printf("2. Add the distances\n");
        printf("3. EXIT\n");
        printf("Enter your option: ");
        scanf("%d", &option);
        printf("\n");

        switch (option) {
            case 1:
                printf("Enter the first distance:\n");
                printf("Kilometers: ");
                scanf("%d", &d1.kms);
                printf("Meters: ");
                scanf("%d", &d1.meters);

                printf("Enter the second distance:\n");
                printf("Kilometers: ");
                scanf("%d", &d2.kms);
                printf("Meters: ");
                scanf("%d", &d2.meters);
```

A function declaration without a p

```c
int main() {
                inputFlag = 1;
                printf("Distances input successfully!\n\n");
                break;

            case 2:
                if (inputFlag) {
                    d3 = addDistance(d1, d2);
                    printf("The sum is %d kms and %d meters\n\n", d3.kms, d3.meters);
                } else {
                    printf("Please input distances first (Option 1)!\n\n");
                }
                break;

            case 3:
                printf("Exiting the program. Goodbye!\n\n");
                break;

            default:
                printf("Invalid option! Please try again.\n\n");
                break;
        }
    } while (option != 3);

    return 0;
}

DISTANCE addDistance(DISTANCE d1, DISTANCE d2) {
    DISTANCE result;
    result.kms = d1.kms + d2.kms;
    result.meters = d1.meters + d2.meters;

    while (result.meters >= 1000) {
        result.meters = result.meters % 1000;
        result.kms += 1;
    }

    return result;
}
```

```
********MAIN MENU**********
1. Input distances
2. Add the distances
3. EXIT
Enter your option: 1

Enter the first distance:
Kilometers: 30
Meters: 21
Enter the second distance:
Kilometers: 34
Meters: 35
Distances input successfully!

********MAIN MENU**********
1. Input distances
2. Add the distances
3. EXIT
Enter your option: 2

The sum is 64 kms and 56 meters

********MAIN MENU**********
1. Input distances
2. Add the distances
3. EXIT
Enter your option: 5

Invalid option! Please try again.

********MAIN MENU**********
1. Input distances
2. Add the distances
3. EXIT
Enter your option: 3

Exiting the program. Goodbye!

Program ended with exit code: 0
```

```c
#include <stdio.h>

typedef struct distance {
    int kms;
    int meters;
} DISTANCE;

DISTANCE addDistance(int kms1, int meters1, int kms2, int meters2);

int main() {
    DISTANCE d1 = {0, 0}, d2 = {0, 0}, d3 = {0, 0};
    int option;
    int inputFlag = 0;

    do {
        printf("********MAIN MENU**********\n");
        printf("1. Input distances\n");
        printf("2. Add the distances\n");
        printf("3. EXIT\n");
        printf("Enter your option: ");
        scanf("%d", &option);
        printf("\n");

        switch (option) {
            case 1:
                printf("Enter the first distance in kms and meters: ");
                scanf("%d %d", &d1.kms, &d1.meters);

                printf("Enter the second distance in kms and meters: ");
                scanf("%d %d", &d2.kms, &d2.meters);
                inputFlag = 1;
                printf("Distances input successfully!\n\n");
                break;

            case 2:
                if (inputFlag) {
```

A function declaration without a prototy

```c
int main() {

            case 2:
                if (inputFlag) {
                    d3 = addDistance(d1.kms, d1.meters, d2.kms, d2.meters);
                    printf("The sum is %d kms and %d meters\n\n", d3.kms, d3.meters);
                } else {
                    printf("Please input distances first (Option 1)!\n\n");
                }
                break;

            case 3:
                printf("Exiting the program. Goodbye!\n\n");
                break;

            default:
                printf("Invalid option! Please try again.\n\n");
                break;
        }
    } while (option != 3);

    return 0;
}

DISTANCE addDistance(int a, int b, int c, int d) {
    DISTANCE result;
    result.kms = a + c;
    result.meters = b + d;

    if (result.meters >= 1000) {
        result.kms += result.meters / 1000;
        result.meters = result.meters % 1000;
    }

    return result;
}
```

```
********MAIN MENU**********
1. Input distances
2. Add the distances
3. EXIT
Enter your option: 4

Invalid option! Please try again.

********MAIN MENU**********
1. Input distances
2. Add the distances
3. EXIT
Enter your option: 2

Please input distances first (Option 1)!

********MAIN MENU**********
1. Input distances
2. Add the distances
3. EXIT
Enter your option: 1

Enter the first distance in kms and meters: 34
23
Enter the second distance in kms and meters: 24
24
Distances input successfully!
```

```
********MAIN MENU**********
1. Input distances
2. Add the distances
3. EXIT
Enter your option: 1

Enter the first distance in kms and meters: 34
23
Enter the second distance in kms and meters: 24
24
Distances input successfully!

********MAIN MENU**********
1. Input distances
2. Add the distances
3. EXIT
Enter your option: 2

The sum is 58 kms and 47 meters

********MAIN MENU**********
1. Input distances
2. Add the distances
3. EXIT
Enter your option: 3

Exiting the program. Goodbye!

Program ended with exit code: 0
```