

The screenshot shows the Visual Studio Code (VS Code) interface. On the left, the Explorer sidebar lists various files and folders, with 'main.c' selected. The main area displays the content of 'main.c', which includes a switch statement mapping days of the week to integer values. Below the editor, the terminal window shows the execution of the program, displaying the output of the switch statement.

```
C main.c M X
C main.c > main()
1 #include <stdio.h>
2
3 int main() {
4
5     enum Days { SUNDAY, MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, SATURDAY };
6
7     enum Days day;
8
9     printf("Days of the week with integer values:\n");
10
11    for (day = SUNDAY; day <= SATURDAY; day++) {
12        switch(day) {
13            case SUNDAY:   printf("SUNDAY = %d\n", day); break;
14            case MONDAY:   printf("MONDAY = %d\n", day); break;
15            case TUESDAY:  printf("TUESDAY = %d\n", day); break;
16            case WEDNESDAY: printf("WEDNESDAY = %d\n", day); break;
17            case THURSDAY: printf("THURSDAY = %d\n", day); break;
18            case FRIDAY:   printf("FRIDAY = %d\n", day); break;
19            case SATURDAY: printf("SATURDAY = %d\n", day); break;
20        }
21    }
22
23    return 0;
24 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
● abhaygupta@Abhays-MacBook-Air-2 main.c % cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
Days of the week with integer values:
SUNDAY = 0
MONDAY = 1
TUESDAY = 2
WEDNESDAY = 3
THURSDAY = 4
FRIDAY = 5
SATURDAY = 6
○ abhaygupta@Abhays-MacBook-Air-2 main.c %
```

The screenshot shows the Visual Studio Code (VS Code) interface. The left sidebar displays the Explorer view with the following file structure:

- OPEN EDITORS:
  - main.c (M)
  - MAIN.C
- MAIN.C
  - .vscode
  - main.dSYM
  - #include <stdio.c
  - a.out
  - file.txt
  - hello.c
  - info.txt
  - input.txt
  - main
  - main.c (M)
  - number.txt
  - numbers.txt
  - output.txt
  - sample.txt
  - students.txt
  - tempCodeRunnerFile
  - tempCodeRunnerFile.c
  - Untitled-1

The main.c file is currently selected and displayed in the central code editor. The code is as follows:

```
C main.c M X
C main.c > main()
1 #include <stdio.h>
2
3 int main() {
4     enum TrafficLight { RED, YELLOW, GREEN };
5
6     enum TrafficLight light;
7
8     for (light = RED; light <= GREEN; light++) {
9         switch(light) {
10            case RED:
11                printf("RED: Stop\n");
12                break;
13            case YELLOW:
14                printf("YELLOW: Wait\n");
15                break;
16            case GREEN:
17                printf("GREEN: Go\n");
18                break;
19        }
20    }
21    return 0;
22 }
```

Below the editor is the Terminal tab, which shows the output of running the program:

```
cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
● abhaygupta@Abhays-MacBook-Air-2 main.c % cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
RED: Stop
YELLOW: Wait
GREEN: Go
○ abhaygupta@Abhays-MacBook-Air-2 main.c %
```

The screenshot shows a Visual Studio Code (VS Code) interface with the following details:

- EXPLORER**: Shows the file tree with the following files:
  - OPEN EDITORS**: `main.c`
  - MAIN.C**:
    - `.vscode`
    - `main.dSYM`
    - `#include <stdio.c`
    - `a.out`
    - `file.txt`
    - `hello.c`
    - `info.txt`
    - `input.txt`
    - `main`
  - `main.c`
  - `number.txt`
  - `numbers.txt`
  - `output.txt`
  - `sample.txt`
  - `students.txt`
  - `tempCodeRunnerFile`
  - `tempCodeRunnerFile.c`
  - `Untitled-1`
- CODE**: The main editor window displays the `main.c` file content:

```
C main.c M X
C main.c > main()
3 int main() {
16     for (month = JANUARY; month <= DECEMBER; month++) {
17         switch(month) {
18             case JANUARY: printf("JANUARY: %d days\n", days[month]); break;
19             case FEBRUARY: printf("FEBRUARY: %d days\n", days[month]); break;
20             case MARCH: printf("MARCH: %d days\n", days[month]); break;
21             case APRIL: printf("APRIL: %d days\n", days[month]); break;
22             case MAY: printf("MAY: %d days\n", days[month]); break;
23             case JUNE: printf("JUNE: %d days\n", days[month]); break;
24             case JULY: printf("JULY: %d days\n", days[month]); break;
25             case AUGUST: printf("AUGUST: %d days\n", days[month]); break;
26             case SEPTEMBER: printf("SEPTEMBER: %d days\n", days[month]); break;
27             case OCTOBER: printf("OCTOBER: %d days\n", days[month]); break;
28             case NOVEMBER: printf("NOVEMBER: %d days\n", days[month]); break;
29             case DECEMBER: printf("DECEMBER: %d days\n", days[month]); break;
30         }
31     }
32     return 0;
33 }
```
- TERMINAL**: The terminal window shows the output of running the program:

```
abhaygupta@Abhay-MacBook-Air-2 main.c % cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c" "main"
Months and their number of days:
JANUARY: 31 days
FEBRUARY: 28 days
MARCH: 31 days
APRIL: 30 days
MAY: 31 days
JUNE: 30 days
JULY: 31 days
AUGUST: 31 days
SEPTEMBER: 30 days
OCTOBER: 31 days
NOVEMBER: 30 days
DECEMBER: 31 days
abhaygupta@Abhay-MacBook-Air-2 main.c %
```

The screenshot shows the Visual Studio Code (VS Code) interface. On the left is the Explorer sidebar with a tree view of files and folders. In the center is the main editor window displaying a C program named 'main.c'. At the bottom is the Terminal pane showing the execution of the program.

**EXPLORER**

- OPEN EDITORS
- MAIN.C
  - .vscode
  - main.dSYM
  - #include <stdio.c
  - a.out
  - file.txt
  - hello.c
  - info.txt
  - input.txt
  - main
  - main.c**
  - number.txt
  - numbers.txt
  - output.txt
  - sample.txt
  - students.txt
  - tempCodeRunnerFile
  - tempCodeRunnerFile.c
  - Untitled-1

**main.c**

```
C main.c M X
C main.c > main()
1 #include <stdio.h>
2
3 int main() {
4     enum Status { SUCCESS, FAILURE, TIMEOUT };
5
6     enum Status currentStatus;
7
8     for (currentStatus = SUCCESS; currentStatus <= TIMEOUT; currentStatus++) {
9         switch(currentStatus) {
10             case SUCCESS:
11                 printf("Status: SUCCESS - Operation completed successfully.\n");
12                 break;
13             case FAILURE:
14                 printf("Status: FAILURE - Operation failed.\n");
15                 break;
16             case TIMEOUT:
17                 printf("Status: TIMEOUT - Operation timed out.\n");
18                 break;
19         }
20     }
21
22     return 0;
23 }
```

**TERMINAL**

```
cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
abhaygupta@Abhay-MacBook-Air-2:~/main % cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
Status: SUCCESS - Operation completed successfully.
Status: FAILURE - Operation failed.
Status: TIMEOUT - Operation timed out.
abhaygupta@Abhay-MacBook-Air-2:~/main %
```

**OUTLINE**

The screenshot shows a Visual Studio Code (VS Code) interface. The left sidebar displays the file tree under the 'OPEN EDITORS' section, with 'main.c' selected. The main editor area shows the following C code:

```
C main.c M X
C main.c > main()
1 #include <stdio.h>
2
3 int main() {
4
5     enum Status { SUCCESS = 10, FAILURE, TIMEOUT };
6
7     printf("SUCCESS = %d\n", SUCCESS);
8     printf("FAILURE = %d\n", FAILURE);
9     printf("TIMEOUT = %d\n", TIMEOUT);
10
11    return 0;
12 }
```

The terminal at the bottom shows the execution of the program:

```
cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
● abhaygupta@Abhay-MacBook-Air-2 main.c % cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
SUCCESS = 10
FAILURE = 11
TIMEOUT = 12
○ abhaygupta@Abhay-MacBook-Air-2 main.c %
```

The screenshot shows a Visual Studio Code (VS Code) interface with the following details:

- EXPLORER** sidebar: Shows a list of files and folders. The file `main.c` is currently selected.
- OPEN EDITORS** sidebar: Shows the same list of files and folders.
- CODE EDITOR**: Displays the content of `main.c`. The code implements a simple menu-based calculator. It includes a `main()` function that prints a menu, takes user input for choice and numbers, and performs addition, subtraction, or multiplication based on the choice.
- TERMINAL**: Shows the output of running the program in the terminal. The user selects option 1 (ADD), enters two numbers (2 and 3), and the result (5.00) is displayed.

```
C main.c M ×
C main.c > main()
1 #include <stdio.h>
2
3 int main() {
4     enum Menu { ADD = 1, SUBTRACT, MULTIPLY };
5
6     int choice;
7     float num1, num2, result;
8
9     printf("Menu:\n");
10    printf("1. ADD\n");
11    printf("2. SUBTRACT\n");
12    printf("3. MULTIPLY\n");
13    printf("Enter your choice: ");
14    scanf("%d", &choice);
15
16    printf("Enter two numbers: ");
17    scanf("%f %f", &num1, &num2);
18
19    switch (choice) {
20        case ADD:
21            result = num1 + num2;
22            printf("Result: %.2f\n", result);
23            break;
24        case SUBTRACT:
25            result = num1 - num2;
}
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
abhaygupta@Abhays-MacBook-Air-2 main.c % cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
Menu:
1. ADD
2. SUBTRACT
3. MULTIPLY
Enter your choice: 1
Enter two numbers: 2
3
Result: 5.00
abhaygupta@Abhays-MacBook-Air-2 main.c %
```

The screenshot shows the Microsoft Visual Studio Code (VS Code) interface. The left sidebar displays the Explorer view, showing files like '.vscode', 'main.dSYM', 'main.c', and various log files. The main area shows the content of 'main.c'. The terminal at the bottom shows the execution of the program.

```
C main.c M X
C main.c > main()
1 #include <stdio.h>
2
3 int main() {
4     enum UserRole { ADMIN = 1, USER, GUEST };
5
6     int choice;
7
8     printf("Select user role:\n");
9     printf("1. ADMIN\n");
10    printf("2. USER\n");
11    printf("3. GUEST\n");
12    printf("Enter your choice: ");
13    scanf("%d", &choice);
14
15    switch(choice) {
16        case ADMIN:
17            printf("Welcome ADMIN! You have full access.\n");
18            break;
19        case USER:
20            printf("Welcome USER! You have limited access.\n");
21            break;
22        case GUEST:
23            printf("Welcome GUEST! You have guest access.\n");
24            break;
25        default:
26    }
}

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
● abhaygupta@Abhays-MacBook-Air-2 main.c % cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
Select user role:
1. ADMIN
2. USER
3. GUEST
Enter your choice: 1
Welcome ADMIN! You have full access.
○ abhaygupta@Abhays-MacBook-Air-2 main.c %
```

The screenshot shows the Visual Studio Code (VS Code) interface. The left sidebar contains icons for file operations like Open, Save, Find, and Delete. The Explorer sidebar lists files and folders, with 'main.c' selected. The main area displays the code for 'main.c':

```
C main.c M X
C main.c > main()
1 #include <stdio.h>
2
3 int main() {
4     enum UserRole { ADMIN = 1, USER, GUEST };
5
6     const char *roleNames[] = {"ADMIN", "USER", "GUEST"};
7
8     for (int i = ADMIN; i <= GUEST; i++) {
9         printf("%s = %d\n", roleNames[i - 1], i);
10    }
11
12 }
13
```

The Terminal tab at the bottom shows the command-line output of running the program:

```
cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
● abhaygupta@192 main.c % cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
ADMIN = 1
USER = 2
GUEST = 3
○ abhaygupta@192 main.c %
```



The screenshot shows the Visual Studio Code (VS Code) interface. On the left is the Explorer sidebar, which lists files and folders. In the center is the main editor area displaying a C program named `main.c`. Below the editor is the Terminal pane, which shows the execution of the program and its output.

**EXPLORER**

- OPEN EDITORS
- `main.c` M
- MAIN.C
- `.vscode`
- `main.dSYM`
- C `#include <stdio.c`
- `a.out`
- `file.txt`
- C `hello.c`
- `info.txt`
- `input.txt`
- `main`
- C `main.c` M
- `number.txt`
- `numbers.txt`
- `output.txt`
- `sample.txt`
- `students.txt`
- C `tempCodeRunnerFile`
- C `tempCodeRunnerFile.c`
- `Untitled-1`

**main.c**

```
C main.c M X
C main.c > main()
11 struct Person {
12 };
13
14 int main() {
15     struct Person p;
16
17     printf("Enter name: ");
18     scanf("%s", p.name);
19
20     printf("Enter gender (1 = Male, 2 = Female, 3 = Other): ");
21     scanf("%d", (int*)&p.gender);
22
23     printf("\n--- Person Details ---\n");
24     printf("Name: %s\n", p.name);
25     printf("Gender: ");
26
27     switch (p.gender) {
28         case MALE:
29             printf("Male\n");
30             break;
31         case FEMALE:
32             printf("Female\n");
33             break;
34         case OTHER:
35             printf("Other\n");
36             break;
37     }
38 }
```

**TERMINAL**

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
abhaygupta@192 main.c % cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
Enter name: abhay
Enter gender (1 = Male, 2 = Female, 3 = Other): 1

--- Person Details ---
Name: abhay
Gender: Male
abhaygupta@192 main.c %
```

The screenshot shows the Visual Studio Code (VS Code) interface. The left sidebar contains icons for file operations like Open, Save, Find, and Delete. The Explorer sidebar lists files and folders, with 'main.c' currently selected. The main editor area displays a C program named 'main.c'. The code defines a struct 'Student' and a function 'main()' that prompts the user for student name, roll number, and marks, then prints them out. The terminal at the bottom shows the execution of the program and its output.

```
#include <stdio.h>
struct Student {
    char name[50];
    int roll_no;
    float marks;
};
int main() {
    struct Student s;
    printf("Enter student name: ");
    scanf("%s", s.name);

    printf("Enter roll number: ");
    scanf("%d", &s.roll_no);

    printf("Enter marks: ");
    scanf("%f", &s.marks);

    printf("\n--- Student Details ---\n");
    printf("Name: %s\n", s.name);
    printf("Roll No: %d\n", s.roll_no);
    printf("Marks: %.2f\n", s.marks);
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
abhaygupta@192 main.c % cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
Enter student name: Abhay
Enter roll number: 12
Enter marks: 89

--- Student Details ---
Name: Abhay
Roll No: 12
Marks: 89.00
abhaygupta@192 main.c %
```

The screenshot shows the Visual Studio Code (VS Code) interface. The left sidebar contains icons for file operations like Open, Save, Find, and Delete. The Explorer sidebar lists files and folders, including a folder named 'MAIN.C' containing '.vscode', 'main.dSYM', 'stdio.c', 'a.out', 'file.txt', 'hello.c', 'info.txt', 'input.txt', 'main', and 'main.c'. The main editor area displays a C program named 'main.c'. The code defines a struct 'Student' and reads details of 5 students from the console. The terminal below the editor shows the output of the program, displaying the details of three students.

```
C main.c M X
C main.c > main()
3 struct Student {
7 };
8
9 int main() {
10     struct Student s[5];
11     int i;
12
13     printf("Enter details of 5 students:\n");
14
15     for(i = 0; i < 5; i++) {
16         printf("\nEnter name: ");
17         scanf("%s", s[i].name);
18
19         printf("Enter roll number: ");
20         scanf("%d", &s[i].roll_no);
21
22         printf("Enter marks: ");
23         scanf("%f", &s[i].marks);
24     }
25
26     printf("\n--- All Student Details ---\n");
27
28     for(i = 0; i < 5; i++) {
29
30 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
--- All Student Details ---
Student 1:
Name: abhay
Roll No: 12
Marks: 89.00
Student 2:
Name: rajat
Roll No: 13
Marks: 78.00
Student 3:
Name: pranshu
Roll No: 14
```

The screenshot shows a Visual Studio Code (VS Code) interface with the following details:

- EXPLORER** sidebar: Shows files in the project. The file `main.c` is currently selected.
- OPEN EDITORS**: One editor is open, showing the content of `main.c`.
- CODE EDITOR**: The main area displays the following C code:

```
#include <stdio.h>
struct Student {
    char name[50];
    int roll_no;
    float marks;
};

int main() {
    struct Student s[5];
    int i, maxIndex = 0;

    printf("Enter details of 5 students:\n");
    for(i = 0; i < 5; i++) {
        printf("\nStudent %d:\n", i + 1);
        printf("Enter name: ");
        scanf("%s", s[i].name);

        printf("Enter roll number: ");
        scanf("%d", &s[i].roll_no);

        printf("Enter marks: ");
        scanf("%f", &s[i].marks);
    }
}
```

- TERMINAL**: The terminal window shows the execution of the program and its output.
- OUTPUT**: The output window shows the results of the program execution.
- PROBLEMS**: No problems are listed.
- PORTS**: No ports are listed.
- STATUS BAR**: Shows the command bar with icons for Code, Search, Find, and others.

The terminal output is as follows:

```
Student 4:
Enter name: pranshu
Enter roll number: 14
Enter marks: 79

Student 5:
Enter name: saurabh
Enter roll number: 15
Enter marks: 67

--- Student with Highest Marks ---
Name: abhay
Roll No: 11
Marks: 90.00
abhaygupta@192 main.c %
```

The screenshot shows the Visual Studio Code (VS Code) interface. The left sidebar contains icons for file operations like Open, Save, Find, and Run. The Explorer sidebar lists files and folders, including a folder named 'MAIN.C' containing 'main.c', 'hello.c', and several text files like 'number.txt', 'numbers.txt', etc. The main editor area displays the code for 'main.c'. The terminal at the bottom shows the execution of the program, prompting for student details and displaying the output.

EXPLORER

OPEN EDITORS

C main.c M X

C main.c > main()

```
1 #include <stdio.h>
2
3 struct Student {
4     char name[50];
5     int roll_no;
6     float marks;
7 }
8
9 void displayStudent(struct Student s) {
10     printf("\n--- Student Details ---\n");
11     printf("Name: %s\n", s.name);
12     printf("Roll No: %d\n", s.roll_no);
13     printf("Marks: %.2f\n", s.marks);
14 }
15
16 int main() {
17     struct Student s1;
18
19     printf("Enter student name: ");
20     scanf("%s", s1.name);
21
22     printf("Enter roll number: ");
23     scanf("%d", &s1.roll_no);
24
25     printf("Enter marks: ");
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
abhaygupta@192 main.c % cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
Enter student name: Abhay
Enter roll number: 12
Enter marks: 90

--- Student Details ---
Name: Abhay
Roll No: 12
Marks: 90.00
abhaygupta@192 main.c %
```

OUTLINE

TIMELINE

The screenshot shows a Visual Studio Code (VS Code) interface with the following details:

- Explorer View:** Shows the file structure. The main.c file is currently selected.
- Code Editor:** Displays the C code for a program that reads student details and prints the top student's information.
- Terminal Output:** Shows the execution of the program, inputting details for 5 students and printing the top student's details.

```
int main() {
    struct Student s[5];
    int i;

    printf("Enter details of 5 students:\n");
    for(i = 0; i < 5; i++) {
        printf("\nStudent %d:\n", i + 1);

        printf("Enter name: ");
        scanf("%s", s[i].name);

        printf("Enter roll number: ");
        scanf("%d", &s[i].roll_no);

        printf("Enter marks: ");
        scanf("%f", &s[i].marks);
    }

    struct Student topper = getTopStudent(s, 5);

    printf("\n--- Top Student Details ---\n");
    printf("Name: %s\n", topper.name);
    printf("Roll No: %d\n", topper.roll_no);
    printf("Marks: %.2f\n", topper.marks);
```

Terminal Output:

```
Student 4:  
Enter name: nikhil  
Enter roll number: 15  
Enter marks: 78  
  
Student 5:  
Enter name: saurabh  
Enter roll number: 16  
Enter marks: 89  
  
--- Top Student Details ---  
Name: abhay  
Roll No: 12  
Marks: 99.00
```

The screenshot shows the Visual Studio Code (VS Code) interface. The left sidebar contains icons for file operations like Open, Save, Find, Run, and Settings. The Explorer sidebar lists files and folders, including a folder named 'MAIN.C' containing '.vscode', 'main.dSYM', 'hello.c', 'info.txt', 'input.txt', 'main.c', and several text files like 'number.txt', 'numbers.txt', etc. The main editor area displays the 'main.c' file with the following code:

```
#include <stdio.h>
#include <time.h>

struct Date {
    int day;
    int month;
    int year;
};

struct Employee {
    char name[50];
    int id;
    float salary;
    struct Date joiningDate;
};

int main() {
    struct Employee e;
    printf("Enter employee name: ");
    scanf("%s", e.name);

    printf("Enter employee ID: ");
    scanf("%d", &e.id);

    printf("Enter salary: ");
    scanf("%f", &e.salary);

    printf("Enter joining date (day month year): ");
    scanf("%d %d %d", &e.joiningDate.day, &e.joiningDate.month, &e.joiningDate.year);
}

--- Employee Details ---
Name: Abhay
ID: 590
Salary: 89000.00
Joining Date: 19—218856888-0001
```

The terminal tab at the bottom shows the command-line output of running the program:

```
cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
abhaygupta@192 main.c % cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
Enter employee name: Abhay
Enter employee ID: 590
Enter salary: 89000
Enter joining date (day month year): 19/11/2007

--- Employee Details ---
Name: Abhay
ID: 590
Salary: 89000.00
Joining Date: 19—218856888-0001
abhaygupta@192 main.c %
```

The screenshot shows the Visual Studio Code (VS Code) interface. On the left is the Explorer sidebar, which lists files in the current workspace. The main area contains an editor window for the file `main.c`, showing C code for reading employee data from a file. Below the editor is a terminal window displaying the execution of the program and its output.

**EXPLORER**

- OPEN EDITORS
- MAIN.C
  - .vscode
  - main.dSYM
  - #include <stdio.h>
  - a.out
  - employee.dat
  - file.txt
  - hello.c
  - info.txt
  - input.txt
  - main
  - main.c
  - number.txt
  - numbers.txt
  - output.txt
  - sample.txt
  - students.txt
  - tempCodeRunnerFile
  - tempCodeRunnerFile.c
  - Untitled-1

**main.c**

```
#include <stdio.h>
struct Employee {
    char name[50];
    int id;
    float salary;
};
int main() {
    struct Employee e1, e2;
    FILE *fp;
    printf("Enter employee name: ");
    scanf("%s", e1.name);

    printf("Enter employee ID: ");
    scanf("%d", &e1.id);

    printf("Enter salary: ");
    scanf("%f", &e1.salary);

    fp = fopen("employee.dat", "wb");
    if(fp == NULL) {
        printf("Error opening file!\n");
        return 1;
    }
```

**TERMINAL**

```
cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
abhaygupta@192 main.c % cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
Enter employee name: Abhay
Enter employee ID: 590
Enter salary: 89000

--- Employee Data Read from File ---
Name: Abhay
ID: 590
Salary: 89000.00
abhaygupta@192 main.c %
```

The screenshot shows the Visual Studio Code (VS Code) interface. On the left is the Explorer sidebar, which lists files in the current workspace. The main area contains an open editor for the file `main.c`, showing the following C code:

```
C main.c M X
C main.c > main()
1 #include <stdio.h>
2 #include <string.h>
3
4 struct Student {
5     char name[50];
6     int roll_no;
7     float marks;
8 };
9
10 int areIdentical(struct Student s1, struct Student s2) {
11     if(strcmp(s1.name, s2.name) != 0)
12         return 0;
13
14     if(s1.roll_no != s2.roll_no)
15         return 0;
16
17     if(s1.marks != s2.marks)
18         return 0;
19
20     return 1;
21 }
22
23 int main() {
24     struct Student a, b;
```

Below the editor, the terminal window displays the following command-line interaction:

```
cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
abhaygupta@192 main.c % cd "/Users/abhaygupta/main.c/" && gcc main.c -o main && "/Users/abhaygupta/main.c/">main
Enter details of Student 1:
Name: Abhay
Roll No: 12
Marks: 89

Enter details of Student 2:
Name: Rajat
Roll No: 13
Marks: 88

The two structures are NOT identical.
abhaygupta@192 main.c %
```