1:

Write a C program to do the following

1. Create an array called **OTHours** to store the number of OT hours worked by five employees.
2. Initialize the **OTHours** array with 20, 22, 25,19, 20.
3. Create another array called **OTrate** to store the OT rate of each employee in order.
4. Input the OT rate of each employee from the keyboard and store in the array.
5. Find the employee who earned highest payment.
6. Display the data stored in **OTHours** array, **OTrate** array and employee who earned highest payment.

Sample output :

OTHours array
20, 22, 25,19, 20

OTrate
200.00, 100.00, 150.00, 300.00, 200.00

employee who earned highest payment : 4

2:

Write a C program to do the following

1. Create an array called **item** to store the prices of three items in the shop.
2. Initialize the item array with 90.00, 20.00, 55.00
3. Create another array called **qtySold** to store the quantities sold from each item.
4. Input the quantities sold from each item from the keyboard and store in the array.
5. 5. Find the item which generate lowest income.
6. Display the data stored in **item** array, **qtySold** array and item which generate least income.

Sample output :

Item array
90.00, 20.00, 55.00

qtySold array
20, 55, 16

Item which generate least income : 3

3:

A game has three rounds and the scores of three rounds for two teams are stored in team1 and team2 arrays respectively.

Write a C program to do the following

1. Create an array called team1 to store the scores of three rounds of team 1.
2. Initialize the team1 array with 8, 3, 7
3. Create another array called team2 to store the scores of three rounds of team 2.
4. Input the scores of three rounds of team 2 from the keyboard and store in the array.
5. Display the winner of each round (team 1/ team 2) and overall winner (team which wins more rounds).
6. Display the data stored in team1 array, team2 array, winner of each round and overall winner.

Sample output :

team1
8, 3, 7

team 2
4, 6, 2

Winners

round 1: team1
round 2: team 2
round 3: team 1

Overall winner team 1

4:

1. Create an array called **correctAnswers** to store the correct answers of 6 true/false questions.
2. Initialize the **correctAnswers** array with T, F, F, T, F, T
3. Create another array called **studentAnswers** to store the student answers of the same exam.
4. Input the student answers of one student from the keyboard and store in the array.
5. Find the number of correct answers for that particular student.
6. Display the data stored in **correctAnswers** array, **studentAnswers** array and number of correct answers for that particular student.

Sample output :

correctAnswers
T. F. F. T. F.T

studentAnswers
T.F.T. T. F. F

number of correct answers : 4

5:

An exam consists of 2 components and 5 students participate for the exam. The marks of both components (out of 100) of all the students are stored in an integer 2D array called **marks**. Each row in the array represents component 1 and component 2 marks of a student. A sample of **marks** array is given below.

30	54
46	55
89	85
90	78
64	73

Write a C program to do the following.

1. Declare an array called **marks** with 5 rows and 2 columns.

2. Read the marks of 5 students and store the marks in the **marks** array.

3. Find the final mark of each student and store the result in another array called **finalMark**.

Final mark = component 1* 40% + component2 * 60%

4. Display the final marks of each student.

6:

Write a C program to do the following

1. Define a structure called **Product** to store the product details Product ID, name, unit price and quantity sold

EX:

A298 | Soap | 65.00 | 40

2. Declare an array of **Product** to store the details of 4 products.

3. Input the details of 4 products from the keyboard and store in the above array.

4. Read the data from the array and display the product ID, name and the amount received from selling each product.

Amount = unit price * quantity

5. Also, total amount received from selling all the products.

Sample output :

Product ID Name Amount
……..
……..
……..
……..
Total ……..

7:

Write a C program to do the following

1. Define a structure called Book to store the details of Books in a library (Book ID, title, no. of copies, number of readers)

EX:

B4500 | Java | 5 | 1278

2. Declare an array of **Book** to store the details of 3 books.

3. Input the of books from the keyboard and store in the above array.

4. Read the data from the array, find and display the most popular book.

Most popular book is the book which has highest number of readers.

Sample Output :

Name of the book : ……
Number of readers : ……

8:

Write a C program to do the following.

1. Define a structure called Employee to store the employee details Employee ID, name, experience (in years) and salary.

EX:

E190 | Niaml | 2 | 35000.00

2. Declare an array of **Employee** to store the details of 3 employees.

3. Input the details of employees from the keyboard and store in the above array.

4. Read the data from the array and display the Employee ID, name and increment.

10% from the salary is given as increment for the employees who has worked more than 2 years.

Sample output :

Employee ID Name Increment
……
……
……
……
……

9:

A chocolate manufacturing company has three machines to produce chocolate balls. 5 chocolate balls from each machine were taken to check the average size of the chocolate balls produce from each machine.

Write a C program to do the following.

1. Declare an array called **size** with 3 rows and 5 columns.

2. Input the size of the chocolate balls from the keyboard and store the sizes in the array called **size**. Assume that each row in the array represent the size of chocolate balls from one machine.

22	22.5	22.3	22.1	21.9
22.6	22.5	22.4	22.2	22.5
22.3	22.1	22.3	22.3	22.4

3. Find the average size of balls of each machine and store the result in another array called **avgSize**.

4. Display the average size of each machine.

10:

A shipping company uses a 2D array to record the length, width, and height of the boxes they shipped. They have recorded the sizes of 4 boxes in an array call boxes

2	2	2
2	4	4
3	4	4
5	5	3

Write a C program to do the following.

1. Declare an array called **boxes** with 4 rows and 3 columns.
2. Input the length, width, and height of 4 boxes and store in the array.
3. Calculate the volume of each box and store in another array called **volume**.

volume = length x height x width

4. Display the volume of each box.