

Dashboard Examinations Lockdown Browser Practice Test it21261350 Thiwanka Kalpage it212613

Not yet answered  
Marked out of 10.00  
Flag question

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
```

Marking Guide

Declaring arrays correctly - 1 mark  
 Initializing the array - 0.5 mark  
 Taking keyboard inputs and store in the array - 1 mark  
 Calculation - 4 marks

Finish attempt ...  
 Time left 0:39:21  
 1

```
#include <stdio.h>
#define size 6

int main (void)
{
    char correctAnswer[size] = {'T', 'F', 'F', 'T', 'F', 'T'};
    char studentAnswer[size];
    int count = 0, i;

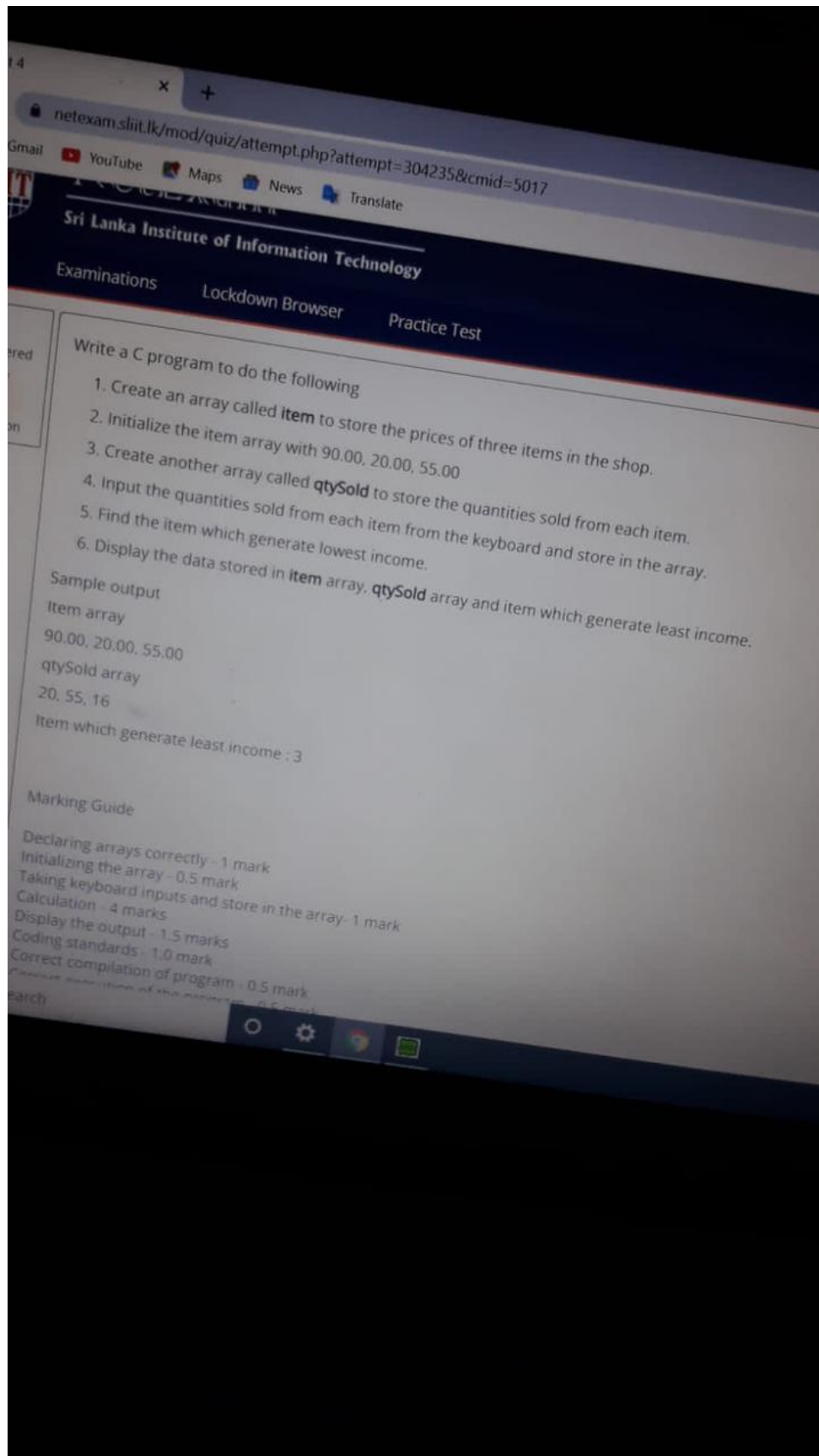
    for (i = 0; i < size; i++)
    {
        printf("%d answer : ", i+1);
        scanf(" %c", &studentAnswer[i]);
    }

    printf("\ncorrectAnswers\n\n");

    for (i = 0; i < size; i++)
    {
        printf("%c ", correctAnswer[i]);
    }

    printf("\n\nstudentAnswers\n\n");
```

```
for (i = 0 ; i < size ; i ++)  
{  
    printf("%c ", studentAnswer[i]);  
  
}  
  
for (i = 0 ; i < size ; i ++)  
{  
    if (correctAnswer[i] == studentAnswer[i] )  
    {  
        count++ ;  
    }  
}  
  
printf("\n\n number of correct Answers : %d" , count);  
  
return 0 ;  
}
```



```

#include <stdio.h>

int main (void)
{
    float item[3] = {90.00 , 20.00 , 55.00} ;
    float least_income = 0;
    int qtySold[3] , i ;
    int total = 0 , count ;

    for (i = 0 ; i < 3 ; i++)
    {
        printf("input item %d sold quantities : " , i+1);
        scanf("%d" , &qtySold[i] );
    }

    printf("item array \n" );

    for (i = 0 ; i < 3 ; i++)
    {
        printf("%.2f , " , item[i]);
    }

    printf("\n\nqtySold array \n" );

    for (i = 0 ; i < 3 ; i++)
    {
        printf("%2d , " , qtySold[i]);
    }

    least_income = item[0] * qtySold[0] ;
    for (i = 0 ; i < 3 ; i++)
    {
        total = qtySold[i] * item[i] ;

        if (total < least_income )
        {
            count = i+1 ;
        }
    }

    printf("\n\nItem which generate least income : %d" , count);
    return 0 ;
}

```

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

Marking Guide

Declaring arrays correctly - 1 mark

Initializing the array - 0.5 mark

Taking keyboard inputs and store in the array- 1 mark

Calculation - 4 marks

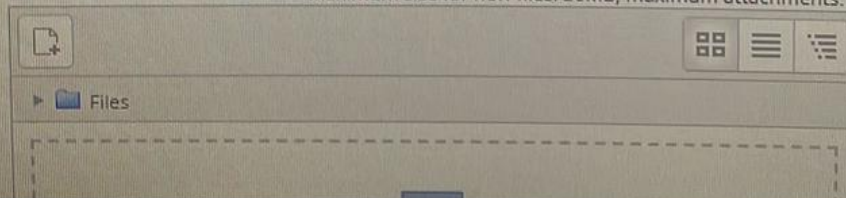
Display the output - 1.5 marks

Coding standards - 1.0 mark

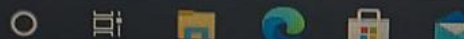
Correct compilation of program - 0.5 mark

Correct execution of the program - 0.5 mark

Maximum size for new files: 30MB, maximum attachments: 1



Click here to search



```

#include <stdio.h>
#define size 5

int main (void)
{
    int OTHours[size] = {20 , 22 , 25 , 19 , 20 } ;
    float OTrate[size] ;
    float h_payment = 0 , payment = 0;
    int count , i ;

    for (i = 0 ; i < size ; i++)
    {
        printf("input the OT rate of %d employee : " , i+1);
        scanf("%f" , &OTrate[i] );
    }

    printf( "\nOTHours array \n\n" );

    for (i = 0 ; i < size ; i++)
    {
        printf("%2d " , OTHours[i]);
    }

    printf("\n\nOTrate \n\n" );

    for (i = 0 ; i < size ; i++)
    {
        printf("%2.2f " , OTrate[i]);
    }
    h_payment = OTHours[0] * OTrate[0] ;

    for (i = 0 ; i < size ; i++)
    {
        payment = OTHours[i] * OTrate[i] ;

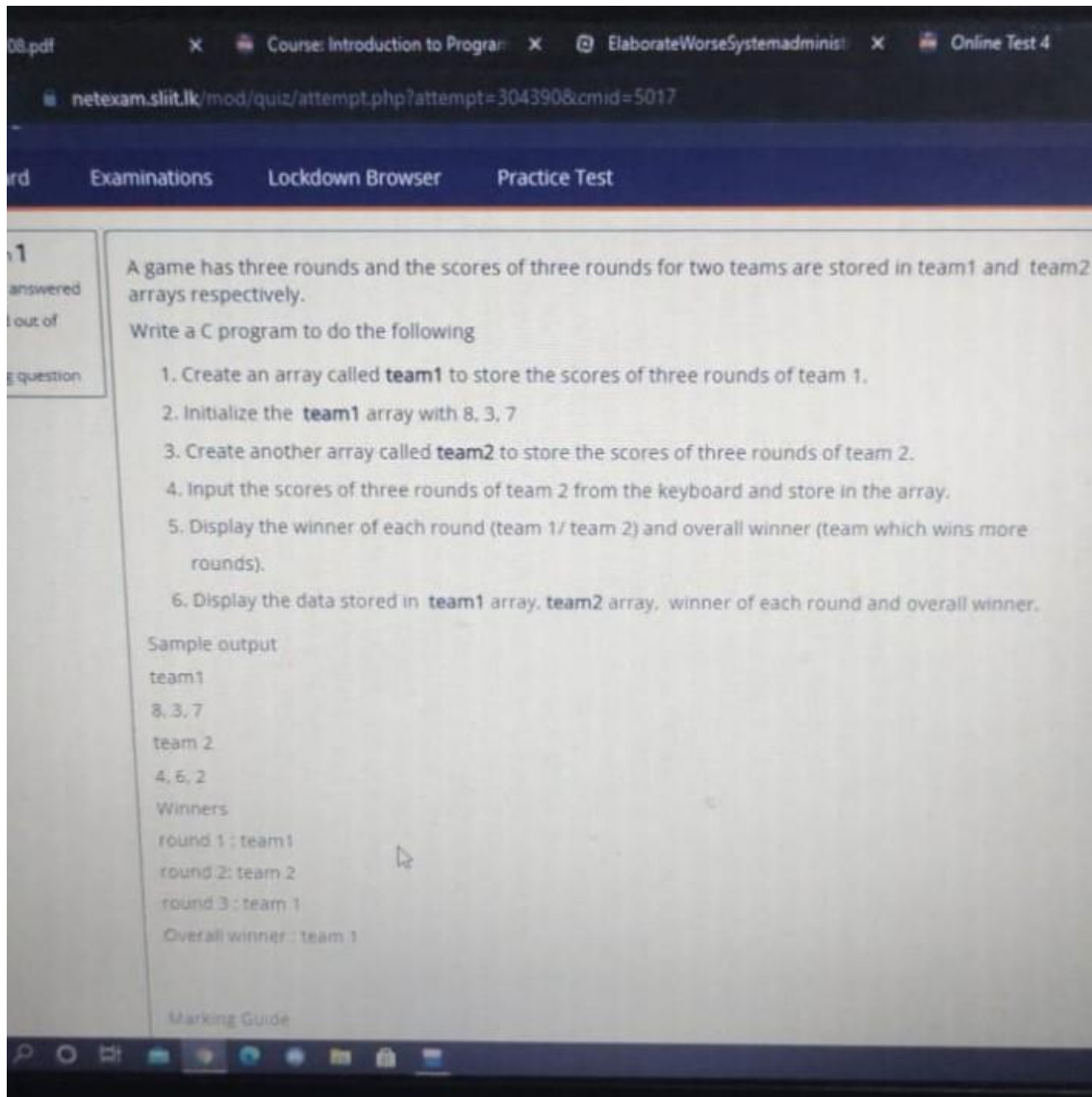
        if (h_payment < payment )
        {
            count = i+1 ;
        }
    }

    printf("\n\nemployee who earned highest payment : %d" , count);
}

```



```
    return 0 ;  
}
```



```
#include <stdio.h>  
#define size 3
```

```
int main (void)  
{  
    int team1[size] = {8 , 3 , 7} ;  
    float least_income = 0;  
    int team2[size] , i ;  
    int total1 = 0 , total2 = 0 ;  
  
    for (i = 0 ; i < size ; i++)  
    {
```

```

        printf("team2 round %d score :", i+1);
        scanf("%d",&team2[i] );
    }

    printf("\nteam 1 \n" );

    for (i = 0 ; i < size ; i++)
    {
        printf("%2d ", team1[i]);
    }

    printf("\n\nteam2 \n" );

    for (i = 0 ; i < size ; i++)
    {
        printf("%2d ", team2[i]);
    }

    printf("\n\nwinners\n\n" );

    for (i = 0 ; i < size ; i++)
    {
        if (team1[i] > team2[i] )
        {
            printf("round %d : team 1\n" , i+1 );
            total1 += team1[i] ;
        }
        else
        {
            printf("round %d : team 2\n" , i+1 );
            total2 += team2[i] ;
        }
    }

    if (total1 > total2 )
    {
        printf("\nOverall winner : team 1" );
    }
    else
    {
        printf("\nOverall winner : team 2" );
    }
    return 0 ;
}

```



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**.  
 $\text{volume} = \text{length} \times \text{height} \times \text{width}$
4. Display the volume of each box.

#### Marking Guide

- Declaring arrays correctly - 0.5 mark
- Taking keyboard inputs and store in the array- 2 marks
- Array Manipulation - 4 marks
- Display the output - 2 marks
- Coding standards - 0.5 mark
- Correct compilation of program - 0.5 mark
- Correct execution of the program - 0.5 mark

Maximum size for new files: 30MB, maximum attachm

```
#include <Stdio.h>
```

```
int main (void)
```

```
{
```

```
    int boxes[4][3] = {0};
```

```
    int volume[3] = {0};
```

```
    int i, j;
```

```
    for (i = 0 ; i < 4 ; i++)
```

```
    {
```

```
        printf("%d box\n" , i+1);
```

```
        printf("input length : ");
```

```
        scanf("%d" , &boxes[i][0]);
```

```
        printf("input Width : ");
```

```
        scanf("%d" , &boxes[i][1]);
```

```

printf("input height : ");
scanf("%d" , &boxes[i][2]);

puts(" ");

}

for (i = 0 ; i < 4 ; i++)
{
    for (j = 0 ; j < 3 ; j++)
    {
        volume[i] += boxes[i][j] ;
    }
    printf("%d box volume : %d\n" , i+1 , volume[i]);
}

return 0 ;
}

```

#### Question 1

Not yet answered  
Marked out of 10.00

Flag question

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 key board 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.

Marking Guide

Declaring arrays correctly - 0.5 mark

Enter the correct code and click on the Submit button.

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

```
int main (void)
```

```
{
```

```
    float size [3][5] = {0};
```

```
    float avgSize[3] = {0} ;
```

```
    int i , j ;
```

```
    for (i = 0 ; i < 3 ; i++)
```

```

{
    printf("%d machine :\n" , i+1);

    for (j = 0 ; j < 5 ; j++)
    {
        printf("size of %d chocolate ball : " , j+1 );
        scanf("%f" , &size[i][j]);
    }
    puts(" ");
}

for (i = 0 ; i < 3 ; i++)
{
    for (j = 0 ; j < 5 ; j++)
    {
        avgSize[i] += size[i][j] ;
    }

    printf("Avarage size of %d machine : %.2f \n" , i+1 , avgSize[i] / 5.0);
}

return 0 ;
}

```

The screenshot shows a web interface for a C programming exam. At the top, there are navigation tabs: Dashboard, Examinations, Lockdown Browser, and Practice Test. The main content area displays 'Question 1' with a status of 'Not yet answered' and a mark of 'Marked out of 10.00'. The question text is: '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. e.g.: E190|Niamh|2|35000.00'. It continues with instructions to declare an array, input details, and calculate a 10% increment for employees with more than 2 years of experience. A sample output table is shown with columns 'Employee ID', 'Name', and 'Increment'. Below the question, a 'Marking Guide' lists criteria and marks: 'Defining the structure correctly - 2.0 marks', 'Declaring arrays correctly - 2.0 mark', 'Input data and store in the array - 1.0', 'Access array elements - 1.0', 'Correct Calculation - 1.5', 'Correct Output - 1.0', and 'Coding standards - 0.5 mark'.

#include <stdio.h>

```

struct Employee
{
    char EmployeeID[20];
    char name[20];
    int exp ;
    float salary ;
};

int main (void)
{
    struct Employee Employee[3] ;
    int i ;
    float increment = 0 ;

    for (i = 0 ; i < 3 ; i++)
    {
        printf("Enter %d Employee ID : " , i+1);
        scanf("%s" , Employee[i].EmployeeID);

        printf("Enter %d Name : " , i+1);
        scanf("%s" , Employee[i].name);

        printf("Enter experince(in year) : " );
        scanf("%d" , &Employee[i].exp);

        printf("Enter %d Salary : " , i+1);
        scanf("%f" , &Employee[i].salary);

        puts (" ");
    }

    printf("\nEmployee ID \t Name \t\t Increment \n");
    for (i = 0 ; i < 3 ; i++)
    {
        if (Employee[i].exp > 2)
        {
            increment = Employee[i].salary / 100 * 10.0 ;
        }
        else
        {
            increment = 0 ;
        }
    }

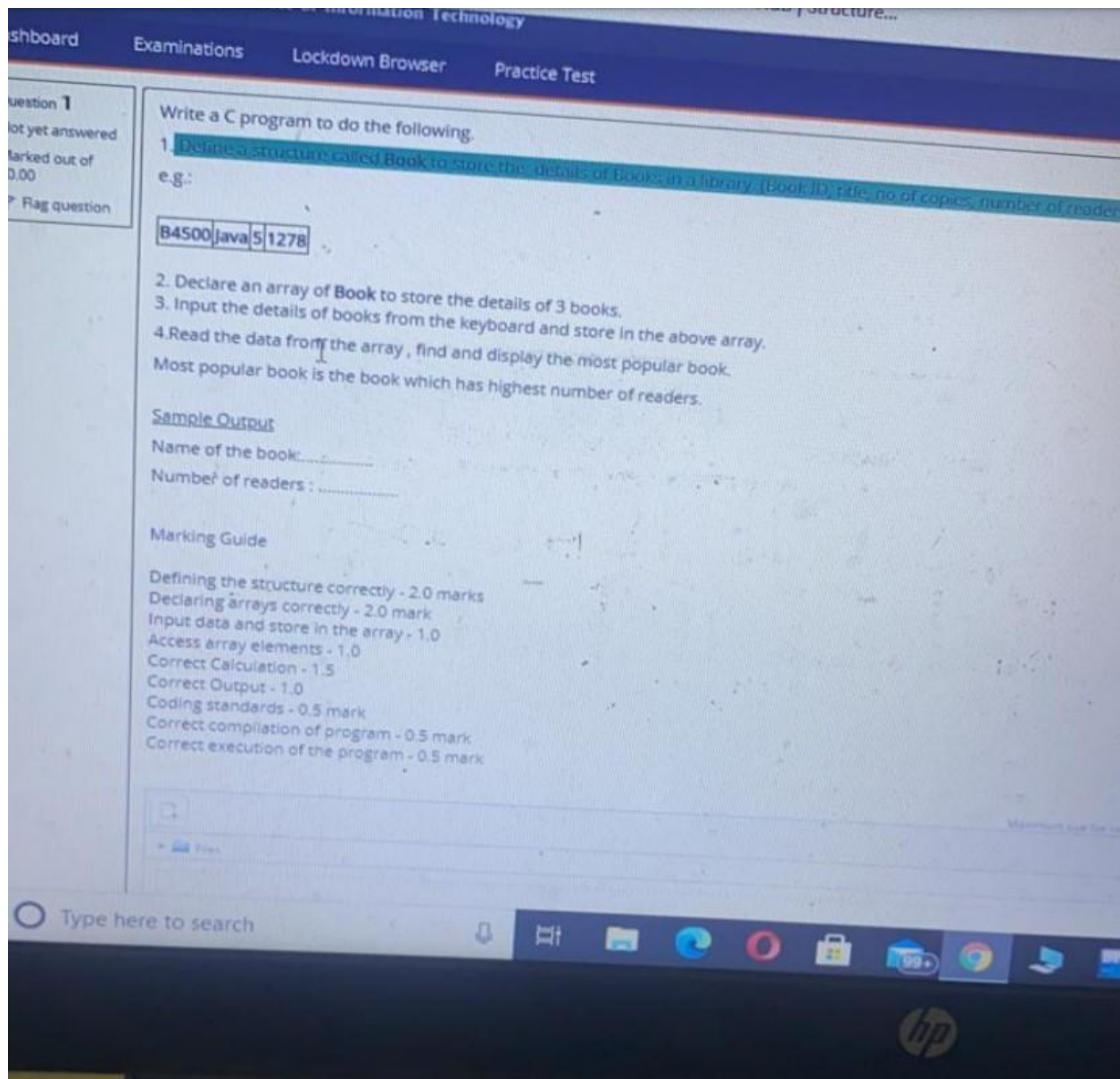
```

```

        printf("%s \t \t %s \t\t %.2f \n" , Employee[i].EmployeeID , Employee[i].name ,
increment );
    }

    return 0;
}

```



```

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

```

```

struct Book
{

```

```

        char BookId[20];
        char title[20] ;
        int copies ;
        int reader ;

    };

int main (void)
{
    struct Book Book[3];
    int n , h , i ;
    char popular[20];

    for (i = 0 ; i < 3 ; i++)
    {
        printf("Enter %d Book ID : " , i+1);
        scanf("%s" , Book[i].BookId);

        printf("Enter %d Book title : " , i+1);
        scanf("%s" , Book[i].title);

        printf("Enter no of copies : " );
        scanf("%d" , &Book[i].copies);

        printf("Enter no of reader : ");
        scanf("%d" , &Book[i].reader);

        puts(" ");
    }
    n = Book[0].reader ;
    for (i = 0 ; i < 3 ; i++)
    {
        if (n < Book[i].reader)
        {
            strcpy(popular , Book[i].title ) ;
            h = Book[i].reader ;
        }
    }
    printf("\nName of the book : %s\n" , popular);
    printf("Nmber of readers : %d" , h);

    return 0;
}

```

[Dashboard](#)
[Examinations](#)
[Lockdown Browser](#)
[Practice Test](#)

it21004018 Mawaththa W.R.Y it21004018

Question 1

Not yet answered

Marked out of 10.00

Flag question

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.

Marking Guide

Declaring arrays correctly - 0.5 mark  
 Taking keyboard inputs and store in the array- 2 marks  
 Array Manipulation - 4 marks  
 Display the output - 2 marks  
 Coding standards - 0.5 mark  
 Correct compilation of program - 0.5 mark  
 Correct execution of the program - 0.5 mark

Quiz navigation

Finish attempt ...

Time left 0:34:55

1

```

#include <stdio.h>
int main (void)
{
    int marks[5][2] = {0};
    int finalmarks[5] = {0} ;
    int i ;

    for (i = 0 ; i < 5 ; i++)
    {
        printf("%d student :\n" , i+1);

        printf("Enter component 1 marks : ");
        scanf("%d" , &marks[i][0]);

        printf("Enter component 2 marks : ");
        scanf("%d" , &marks[i][1]);

        puts(" ");
    }

    for (i = 0 ; i < 5 ; i++)
    {
        finalmarks[i] = marks[i][0] * 40 / 100 + marks[i][1] * 60 / 100 ;

        printf("final marks of %d student : %d \n" , i+1 , finalmarks[i]);
    }
}

```



```

    }

    return 0 ;
}

```

Question 1  
Not yet answered  
Marked out of 10.00  
Flag question

A researcher wants to record the rainfall of 4 cities in Sri Lanka for 3 days. He uses a 2D array for this purpose. Sample of a rainfall array is given below.

48	110	98	89
56	67	78	62
49	59	34	68

Write a C program to do the following.

1. Declare a 2D array called **rainfall** with 3 rows and 4 columns.
2. Input the rainfall of 4 cities in Sri Lanka for 3 days and store in the array.
3. Find the maximum rainfall of each day and store those in another array **maxRainfall**.
4. Display the maximum rainfall of each day.

Marking Guide

- Declaring arrays correctly - 0.5 mark
- Taking keyboard inputs and store in the array- 2 marks
- Array Manipulation - 4 marks
- Display the output - 2 marks
- Coding standards - 0.5 mark
- Correct compilation of program - 0.5 mark
- Correct execution of the program - 0.5 mark

Quiz navigation

Finish attempt ...

Time left 0:34:51

1

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

```
int main (void)
```

```
{
```

```
    int rainfall[3][4];
```

```
    int i , j , max ;
```

```
    int maxrainfall[3];
```

```
    for (i = 0 ; i < 3 ; i++)
```

```
    {
```

```
        printf("%d day :\n" , i+1);
```

```
        for (j = 0 ; j < 4 ; j++)
```

```
        {
```

```

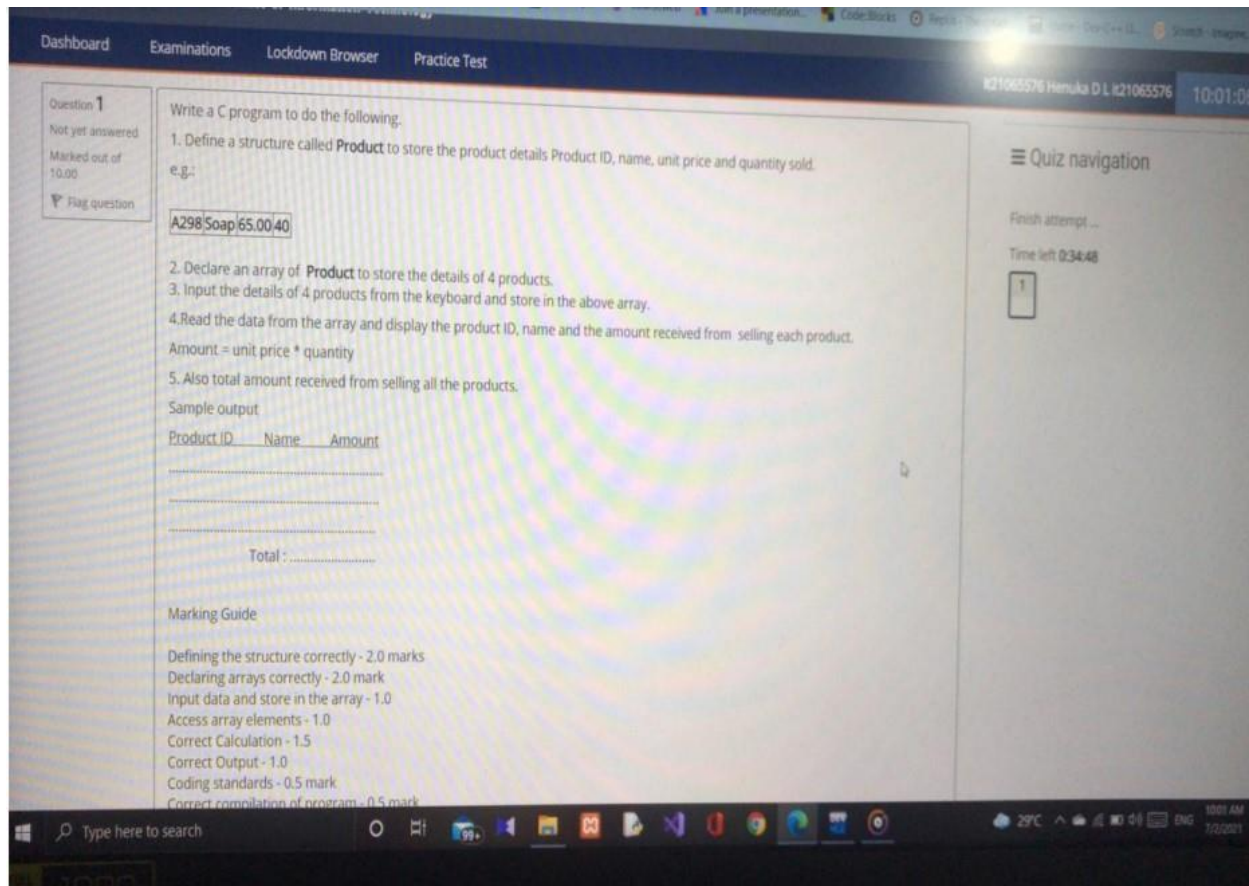
        printf("%d city rainfall : " , j+1);
        scanf("%d" , &rainfall[i][j] );
    }
    puts(" ");
}

for (i = 0 ; i < 3 ; i++)
{
    max = rainfall[i][0];

    for (j = 0 ; j < 4 ; j++)
    {
        if(max <= rainfall[i][j])
        {
            maxrainfall[i] = rainfall[i][j] ;
        }
    }
    printf("\n%d day Maximun rainfall is : %d" , i+1 , maxrainfall[i]);
}

return 0 ;
}

```



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

```
struct Product
{
    char ProductID[20];
    char name[20];
    int quantity ;
    float unitPrice ;
};
```

```
int main (void)
{
    struct Product Product[4] ;
    int i ;
    float amount = 0 , total = 0 ;

    for (i = 0 ; i < 4 ; i++)

    {
        printf("Enter %d Product ID : " , i+1);
```

```

scanf("%s" , Product[i].ProductID);

printf("Enter %d Product Name : " , i+1);
scanf("%s" , Product[i].name);

printf("Enter %d Quantity sold : " , i+1);
scanf("%d" , &Product[i].quantity);

printf("Enter %d Product unit price : " , i+1);
scanf("%f" , &Product[i].unitPrice);

puts (" ");
}

printf("\nProduct ID \t Name \t\t Amount \n");
for (i = 0 ; i < 4 ; i++)
{
    amount = Product[i].unitPrice * Product[i].quantity ;
    total += amount ;
    printf("%s \t \t %s \t\t %.2f \n" ,Product[i].ProductID , Product[i].name , amount );

}
printf("\t \t\t total : %.2f" , total);
return 0;
}

```

Question 1  
Not yet answered  
Marked out of 10.00  
Flag question

The daily temperature of 2 main cities in Sri Lanka is recorded three times a day and stored in a 2D array called **temp**.  
At the end of each day, the average temperature of each city is calculated and stored in another 1D array called **avgTemp**.

Write a C program to do the following.

1. Declare an array called **temp** with 2 rows and 3 columns.
2. Input the temperatures from the key board and store in the array.
3. Calculate the average temperature of each city and store the result in **avgTemp** array in the same order of cities.
4. Display the average temperatures of the cities.

example :

temp array			
	morning	noon	evening
Colombo	29.1	32.6	31.3
Kandy	27.8	30.2	28.4

avgTemp array	
Colombo	31.0
Kandy	28.8

Marking Guide

- Declaring arrays correctly - 0.5 mark
- Taking keyboard inputs and store in the array - 2 marks
- Array Manipulation - 4 marks
- Display the output - 2 marks
- Coding standards - 0.5 mark
- Correct compilation of program - 0.5 mark
- Correct execution of the program - 0.5 mark

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

```
int main (void)
```

```
{
```

```
    float temp[2][3] = {0};
```

```
    float avgTemp[2] = {0};
```

```
    int i , j ;
```

```
    for (i = 0 ; i < 2 ; i++)
```

```
    {
```

```
        printf("%d city :\n", i+1);
```

```
        for (j = 0 ; j < 3 ; j++)
```

```
        {
```

```
            printf("%d time temperature on the day : " , j+1);
```

```
            scanf("%f" , &temp[i][j]);
```

```
        }
```

```

        puts(" ");
    }

    for (i = 0 ; i < 2 ; i++)
    {
        for (j = 0 ; j < 3 ; j++)
        {
            avgTemp[i] += temp[i][j] ;
        }
        printf("%d city average temperature of whole day : %.2f\n", i+1 , avgTemp[i] / 3.0)
    ;
    }

    return 0 ;
}

```