

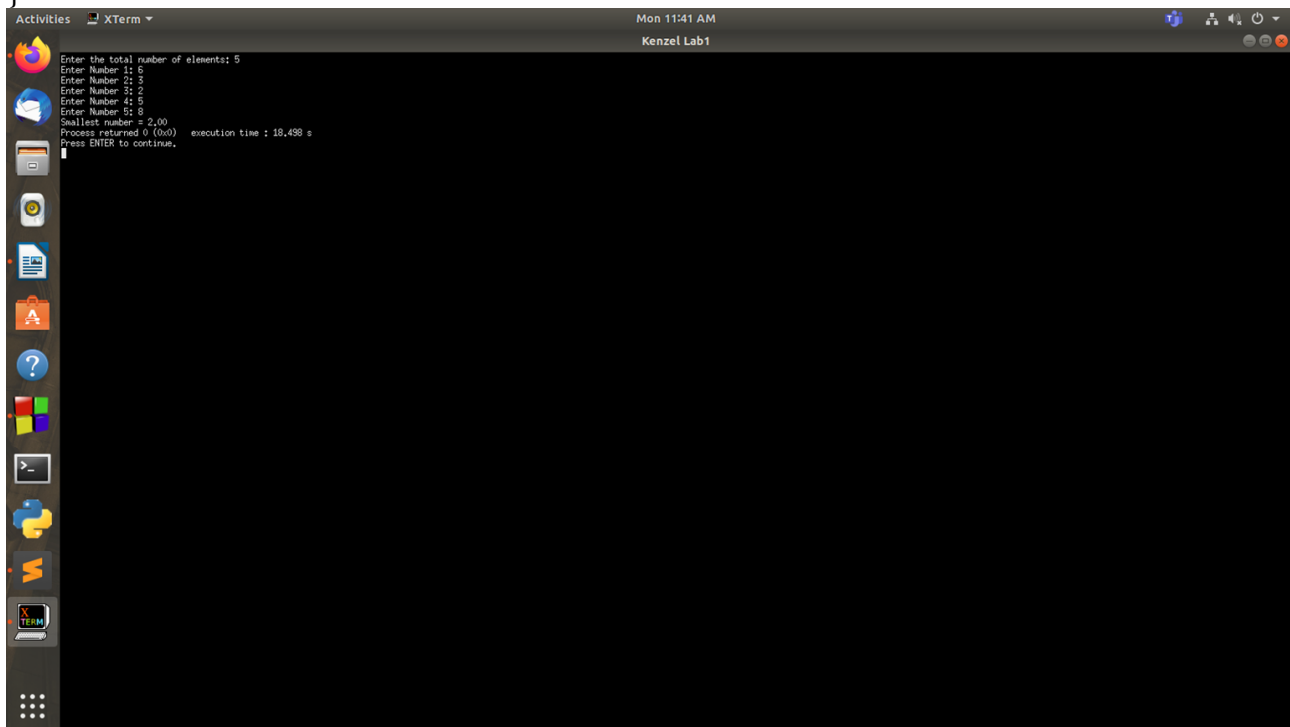
Lab 1

Q1

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int num;
    float *data;
    printf("Enter the total number of elements: ");
    scanf("%d", &num);

    for (int i = 0; i < num; ++i) {
        printf("Enter Number %d: ", i + 1);
        scanf("%f", data + i);
    }
    for (int i = 1; i < num; ++i) {
        if (*data > *(data + i))
            *data = *(data + i);
    }
    printf("Smallest number = %.2f", *data);

    return 0;
}
```



The screenshot shows a terminal window titled 'XTerm' with the following output:

```
Enter the total number of elements: 5
Enter Number 1: 8
Enter Number 2: 3
Enter Number 3: 2
Enter Number 4: 5
Enter Number 5: 9
Smallest number = 2.00
Process returned 0 (0x0)   execution time : 18.498 s
Press ENTER to continue.
```

Q2

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

```
#define ROWS 3
```

```
#define COLS 3
```

```
void matrixInput(int mat[][COLS]);
```

```
void matrixPrint(int mat[][COLS]);
```

```
void matrixMulti(int mat1[][COLS], int mat2[][COLS], int res[][COLS]);
```

```
int main()
```

```
{
```

```
    int mat1[ROWS][COLS], mat2[ROWS][COLS], res[ROWS][COLS];
```

```
    printf("Enter elements in first matrix of size %dx%d: \n", ROWS, COLS);
```

```
    matrixInput(mat1);
```

```
    printf("\nEnter elemetns in second matrix of size %dx%d: \n", ROWS, COLS);
```

```
    matrixInput(mat2);
```

```
    matrixMulti(mat1, mat2, res);
```

```
    printf("\nProduct of first and second matrix: \n");
```

```
    matrixPrint(res);
```

```
    return 0;
```

```
}
```

```
void matrixInput(int mat[][COLS])
```

```
{
```

```
    int i, j;
```

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

```
    {
```

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

```
        {
```

```
            scanf("%d", (*(mat + i) + j));
```

```
        }
```

```
    }
```

```
}
```

```

void matrixPrint(int mat[][COLS])
{
    int i, j;

    for (i = 0; i < ROWS; i++)
    {
        for (j = 0; j < COLS; j++)
        {
            printf("%d ", *(mat + i) + j));
        }
        printf("\n");
    }
}

```

```

void matrixMulti(int mat1[][COLS], int mat2[][COLS], int res[][COLS])
{
    int i, j, k;
    for(i=0; i<ROWS; i++)
    {
        for(j=0; j<COLS; j++)
        {
            res[i][j]=0;
            for(k=0; k<COLS; k++)
            {
                (*(res + i) + j) += (*(mat1 + i) + k) * (*(mat2 + k) + j);
            }
        }
    }
}

```

The screenshot shows a terminal window titled 'XTerm' with the following output:

```

Enter elements in first matrix of size 3x3:
1 2 3
4 5 6
7 8 9
Enter elements in second matrix of size 3x3:
1 2 3
4 5 6
7 8 9
Product of first and second matrix:
6 12 18
15 22 30
24 32 42
Process returned 0 (0x0)   execution time : 20,641 s
Press ENTER to continue.

```

Q3

```
#include<stdio.h>
struct DOB
{
    int day;
    int month;
    int year;
};
struct ADRS
{
    int house_no;
    long zipcode;
    char state[20];
};
struct EMPLOYEE
{
    char name[20];
    struct DOB dob;
    struct ADRS address;
};
int main()
{
    struct EMPLOYEE emp[10];
    struct EMPLOYEE * ptr;
    int i,n;
    printf("Enter number of Employees\n");
    scanf("%d",&n);
    for(i=0;i<n;i++)
    {
        ptr=emp+i;
        printf("Enter %d Employee Name\n",i+1);
        scanf("%s",ptr->name);
        printf("Enter date of birth\n");
        scanf("%d",&(ptr->dob.day));
        printf("Enter month of birth\n");
        scanf("%d",&(ptr->dob.month));
        printf("Enter year of birth\n");
        scanf("%d",&(ptr->dob.year));
        printf("Enter House Number\n");
        scanf("%d",&(ptr->address.house_no));
        printf("Enter Zip Code\n");
        scanf("%ld",&(ptr->address.zipcode));
        printf("Enter State\n");
        scanf("%s",ptr->address.state);
    }
}
```

```

printf("\n");
for(i=0;i<n;i++)
{
    ptr=emp+i;
    printf("%d Employee's Name is %s\n",i+1,ptr->name);
    printf("Date of birth is %d/%d/%d\n",ptr->dob.day,ptr->dob.month,ptr->dob.year);
    printf("House Number is %d\n",ptr->address.house_no);
    printf("Zip Code is %ld\n",ptr->address.zipcode);
    printf("State is %s\n",ptr->address.state);
    printf("\n");
}
}

```

OnlineGDB beta
online compiler and debugger for c/c++
code. compile. run. debug. share.

IDE
My Projects
Classroom **new**
Learn Programming
Programming Questions
We are Hiring
Sign Up
Login

input

```

Enter year of birth
20
Enter House Number
46
Enter Zip Code
131
Enter State
rte
Enter 2 Employee Name
kin
Enter date of birth
5
Enter month of birth
6
Enter year of birth
85
Enter House Number
45
Enter Zip Code
5213
Enter State
arti

1 Employee's Name is bib
Date of birth is 2/2/20
House Number is 46
Zip Code is 131
State is rte

2 Employee's Name is kin
Date of birth is 5/6/85
House Number is 45
Zip Code is 5213
State is arti

```

About • FAQ • Blog • Terms of Use • Contact Us • ODB
Tutorial • Credits • Privacy
© 2016 - 2021 GDB Online