

DSA LAB

UDEET MITTAL
CSE C SECTION
ROLL NUMBER 64

SOLVED EXCERCISES

1)Write a program to read n names of different sports and store them using array pointers. Use dynamic memory allocation and deallocation functions. The program should display all the names and deallocate the dynamic memory at the end of the program.

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
int main(){
int i,n;
char *sports[10];
char str[100];
printf("Name: Udeet Mittal\nRoll Number:64\nBatch:C3\n");
printf("\n enter the number of sports \n");
scanf("%d", &n);
printf("\nenter the names of sports:\n");
for (i = 0; i < n; i++){
scanf("%s", str);
//allocating memory equal to the length of string + 1
//Last 1 byte to accommodate the '\0'
sports[i] = (char*) calloc(strlen(str)+1, sizeof(char));
strcpy(sports[i],str);
}
printf("\nGiven list of sports: \n");
for (i = 0; i < n; i++)
printf("%s\n", sports[i]);
//Deallocate the dynamic memory
for (i = 0; i < n; i++)
free(sports[i]);
return 0;
}
```

```
Terminal
File Edit View Search Terminal Help
$ gcc q1.c
$ ./a.out
Name: Udeet Mittal
Roll Number:64
Batch:C3

enter the number of sports
3

enter the names of sports:
cricket
hockey
football

Given list of sports:
cricket
hockey
football
$
```

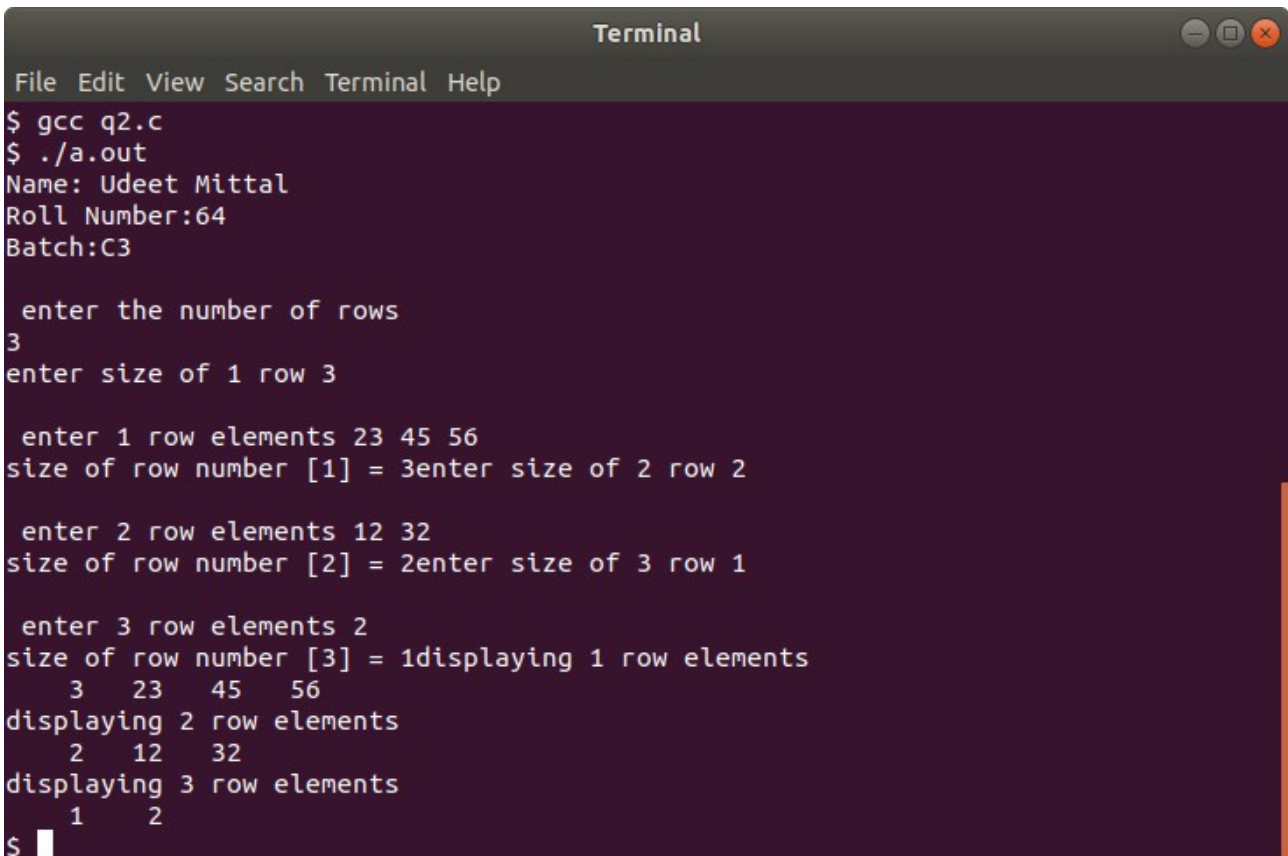
2.) Write a C program to implement a ragged array dynamically.

```
#include<stdio.h>
#include<stdlib.h>
int main()
{
int rowNum, colNum, i, j;
int **table;
printf("Name: Udeet Mittal\nRoll Number:64\nBatch:C3\n");
printf("\n enter the number of rows \n");
scanf("%d", &rowNum);
table = (int **) calloc(rowNum+1, sizeof(int *));
for (i = 0; i < rowNum; i++) /* this will tell which row we are in */
{
printf("enter size of %d row", i+1);
scanf("%d", &colNum);table[i] = (int *) calloc(colNum+1, sizeof(int));
printf("\n enter %d row elements ", i+1);
for (j = 1; j <= colNum; j++)
{
scanf("%d", &table[i][j]);
}
table[i][0] = colNum;
printf("size of row number [%d] = %d", i+1, table[i][0]);
}
table[i] = NULL;
for (i = 0; i < rowNum; i++)
```

```

/* this will tell which row we are in */
{
    printf("displaying %d row elements\n", i+1);
    for (j = 0; j <= *table[i]; j++)
    {
        printf("%5d", table[i][j]);
    }
    printf("\n");
} //freeup the memory
for (i = 0; i < rowNum; i++)
{
    free(table[i]);
}
free(table);
return 0;
}

```



```

Terminal
File Edit View Search Terminal Help
$ gcc q2.c
$ ./a.out
Name: Udeet Mittal
Roll Number:64
Batch:C3

enter the number of rows
3
enter size of 1 row 3

enter 1 row elements 23 45 56
size of row number [1] = 3enter size of 2 row 2

enter 2 row elements 12 32
size of row number [2] = 2enter size of 3 row 1

enter 3 row elements 2
size of row number [3] = 1displaying 1 row elements
    3    23    45    56
displaying 2 row elements
    2    12    32
displaying 3 row elements
    1     2
$ 

```

Questions for LAB1

1. Write a function Smallest to find the smallest element in an array using pointer. Create a dynamically allocated array and read the values from keyboard in main. Display the result in the main function.

Filename: "Smallest.h"

```
int Smallest(int arr[],int n)
{
    int *min=arr;
    for (int i = 0; i < n; ++i)
    {
        if(*min>*(arr+i))
            *min=*(arr+i);
    }
    return *min;
}
```

Filename: "lab1_Q1.c"

```
#include <stdio.h>
#include <stdlib.h>
#include "Smallest.h"

int main()
{
    int i,n;
    int *a;
    printf("Name: Udeet Mittal\nRoll Number:64\nBatch:C3\n");
    printf("Number of elements to be entered=");
    scanf("%d",&n);
    a=(int*)calloc(n,sizeof(int));
    printf("Enter %d numbers:\n",n);
    for(i=0;i<n;i++)
    {
        scanf("%d",&a[i]);
    }
    int min=Smallest(a,n);
    printf("The Smallest Number is: %d \n",min);
    free(a);
    return 0;
}
```

```
Terminal
File Edit View Search Terminal Help
$ gcc lab1_Q1.c
$ ./a.out
Name: Udeet Mittal
Roll Number:64
Batch:C3
Number of elements to be entered=5
Enter 5 numbers:
23
454
67
12
99
The Smallest Number is: 12
$
```

2.Implement a C program to read, display and to find the product of two matrices using functions with suitable parameters.Note that the matrices should be created using dynamic memory allocation functions and the elements are accessed using array dereferencing.

Filename: "Multiply.h"

```
int ** Multiply(int **m1, int **m2, int n, int m, int x){
    int **m3;
    m3 = (int **)calloc(n, sizeof(int*));
    for (int i = 0; i < m; ++i)
    {
        m3[i] = (int *)calloc(m, sizeof(int*));
    }
    for (int i = 0; i < n; ++i)
    {
        for (int j = 0; j < m; ++j)
        {
            for (int k = 0; k < x; ++k)
```

```

        {
            m3[i][j] = m3[i][j] + m1[i][k] * m2[k][j];
        }
    }
}
return m3;
}

```

Filename: "lab1_Q2.c"

```

#include <stdio.h>
#include <stdlib.h>
#include "Multiply.h"
int main()
{
    int **mat1;
    int **mat2;
    int **mat3;
    int n1, m1, n2, m2;
    printf("Name: Udeet Mittal\nRoll Number:64\nBatch:C3\n");
    printf("Enter dimensions of Matrix 1: ");
    scanf("%d", &n1);
    scanf("%d", &m1);
    printf("Enter dimensions of Matrix 2: ");
    scanf("%d", &n2);
    scanf("%d", &m2);
    if(m1==n2)
    {
        mat1 = (int **)calloc(n1, sizeof(int*));
        for (int i = 0; i < m1; ++i)
        {
            mat1[i] = (int *)calloc(m1, sizeof(int*));
        }
        mat2 = (int **)calloc(n2, sizeof(int*));
        for (int i = 0; i < m2; ++i)
        {
            mat2[i] = (int *)calloc(m2, sizeof(int*));
        }
        mat3 = (int **)calloc(n1, sizeof(int*));
        for (int i = 0; i < m2; ++i)
        {
            mat3[i] = (int *)calloc(m2, sizeof(int*));
        }

        printf("Enter Matrix 2:\n");
        for (int i = 0; i < n1; ++i)
        {
            for (int j = 0; j < m1; ++j)
            {

```

```

        printf("[%d][%d]: ", i, j);
        scanf("%d", &mat1[i][j]);
    }

}

printf("Enter Matrix 2:\n");
for (int i = 0; i < n2; ++i)
{
    for (int j = 0; j < m2; ++j)
    {
        printf("[%d][%d]: ", i, j);
        scanf("%d", &mat2[i][j]);
    }
}

printf("Matrix 1:\n");
for (int i = 0; i < n1; ++i)
{
    for (int j = 0; j < m1; ++j)
    {
        printf("%d\t", mat1[i][j]);
    }
    printf("\n");
}

printf("Matrix 2:\n");
for (int i = 0; i < n2; ++i)
{
    for (int j = 0; j < m2; ++j)
    {
        printf("%d\t", mat2[i][j]);
    }
    printf("\n");
}

mat3 = Multiply(mat1, mat2, n1, m2, m1);
printf("Matrix Multiplication:\n");
for (int i = 0; i < n1; ++i)
{
    for (int j = 0; j < m2; ++j)
    {
        printf("%d\t", mat3[i][j]);
    }
    printf("\n");
}
}
return 0;
}

```

```
Terminal
File Edit View Search Terminal Help

$ gcc lab1_Q2.c
$ ./a.out
Name: Udeet Mittal
Roll Number:64
Batch:C3
Enter dimensions of Matrix 1: 2
2
Enter dimensions of Matrix 2: 2
2
Enter Matrix 1:
[0][0]: 1
[0][1]: 2
[1][0]: 3
[1][1]: 4
Enter Matrix 2:
[0][0]: 1
[0][1]: 2
[1][0]: 3
[1][1]: 4
Matrix 1:
1      2
3      4
Matrix 2:
1      2
3      4
Matrix Multiplication:
7      10
15     22
$
```

3.Samuel wants to store the data of his employees, which includes the following fields: (i) Name of the employee (ii) Date of birth which is a collection of {day, month, year} (iii) Address which is a collection of {house number, zip code and state}. Write a 'C' program to read and display the data of N employees using pointers to array of structures.

Filename: "employee.h"

```
struct DOB {
    int day, month, year;
};
```

```
struct ADRS {
    int house_no;
    long zipcode;
    char state[20];
};
```



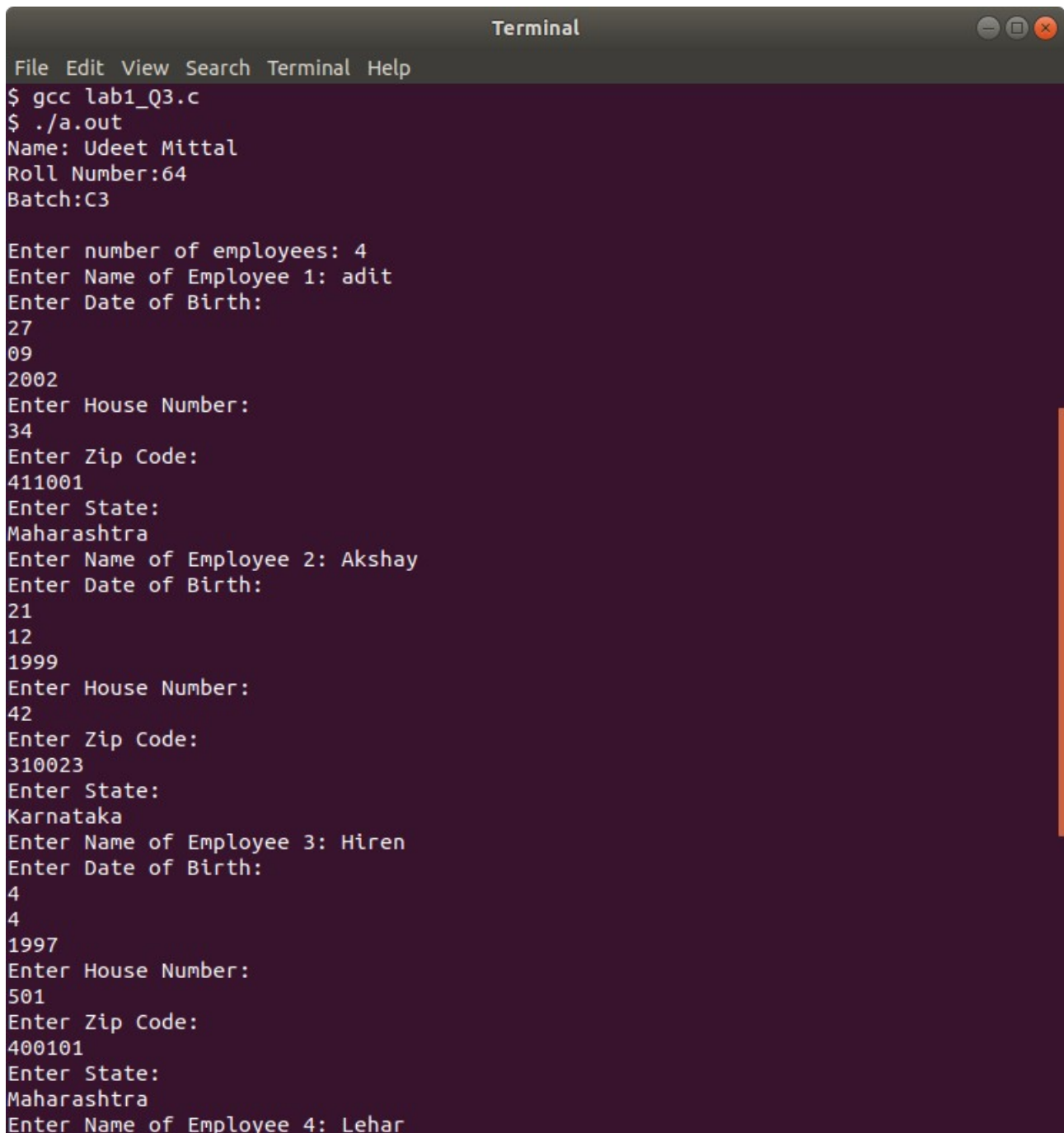
```
struct EMPLOYEE {  
    char name[20];  
    struct DOB dob;  
    struct ADRS address;  
};
```

Filename: "lab1_Q3.c"

```
#include <stdio.h>  
#include <stdlib.h>  
#include "employee.h"  
  
int main()  
{  
    struct EMPLOYEE emp[10];  
    struct EMPLOYEE* ptr = emp;  
    int N;  
    printf("Name: Udeet Mittal\nRoll Number:64\nBatch:C3\n");  
    printf("\nEnter number of employees: ");  
    scanf("%d", &N);  
  
    for (int i = 0; i < N; ++i)  
    {  
        // char n[20];  
        printf("Enter Name of Employee %d: ", (i+1));  
        scanf("%s", (ptr+i)->name);  
        int d, m, y;  
        printf("Enter Date of Birth: \n");  
        scanf("%d %d %d", &d, &m, &y);  
        (ptr+i)->dob.day = d;  
        (ptr+i)->dob.month = m;  
        (ptr+i)->dob.year = y;  
        int hno;  
        long z;  
        char st[20];  
        printf("Enter House Number: \n");  
        scanf("%d", &hno);  
        printf("Enter Zip Code: \n");  
        scanf("%ld", &z);  
        printf("Enter State: \n");  
        scanf("%s", (ptr+i)->address.state);  
        (ptr+i)->address.house_no = hno;  
        (ptr+i)->address.zipcode = z;  
    }  
}
```

```
for (int i = 0; i < N; ++i)
{
    printf("Employee %d\n", (i+1));
    printf("Name: %s\n", (ptr+i)->name);
    printf("Date of Birth: %d-%d-%d\n", (ptr+i)->dob.day, (ptr+i)->dob.month,
(ptr+i)->dob.year);
    printf("Address: %d, %ld, %s\n", (ptr+i)->address.house_no,
(ptr+i)->address.zipcode, (ptr+i)->address.state);

}
return 0;
}
```



```
Terminal
File Edit View Search Terminal Help
$ gcc lab1_Q3.c
$ ./a.out
Name: Udeet Mittal
Roll Number:64
Batch:C3

Enter number of employees: 4
Enter Name of Employee 1: adit
Enter Date of Birth:
27
09
2002
Enter House Number:
34
Enter Zip Code:
411001
Enter State:
Maharashtra
Enter Name of Employee 2: Akshay
Enter Date of Birth:
21
12
1999
Enter House Number:
42
Enter Zip Code:
310023
Enter State:
Karnataka
Enter Name of Employee 3: Hiren
Enter Date of Birth:
4
4
1997
Enter House Number:
501
Enter Zip Code:
400101
Enter State:
Maharashtra
Enter Name of Employee 4: Lehar
```

```
Terminal
File Edit View Search Terminal Help
Enter Name of Employee 4: Lehar
Enter Date of Birth:
23
08
2003
Enter House Number:
45
Enter Zip Code:
231001
Enter State:
Haryana
Employee 1
Name: adit
Date of Birth: 27-9-2002
Address: 34, 411001, Maharashtra
Employee 2
Name: Akshay
Date of Birth: 21-12-1999
Address: 42, 310023, Karnataka
Employee 3
Name: Hiren
Date of Birth: 4-4-1997
Address: 501, 400101, Maharashtra
Employee 4
Name: Lehar
Date of Birth: 23-8-2003
Address: 45, 231001, Haryana
$
```