

DSA Lab

LAB 1 Submission

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.

Program:

```
#include<stdio.h>
#include<stdlib.h>
int
Smallest (int *a, int n)
{

    int *pLast = a + n - 1, *pWalk, smallest = *a;
    for (pWalk = a; pWalk <= pLast; pWalk++)
    {
        if (smallest > *pWalk)
        {
            smallest = *pWalk;
        }
    }

    return smallest;
}

int
main ()
{

    int *a, i, n;
    printf ("Enter the number of elements in the array:\n");
    scanf ("%d", &n);
    printf ("Enter %d elements now: \n", n);
    a = (int *) calloc (n, sizeof (int));
```

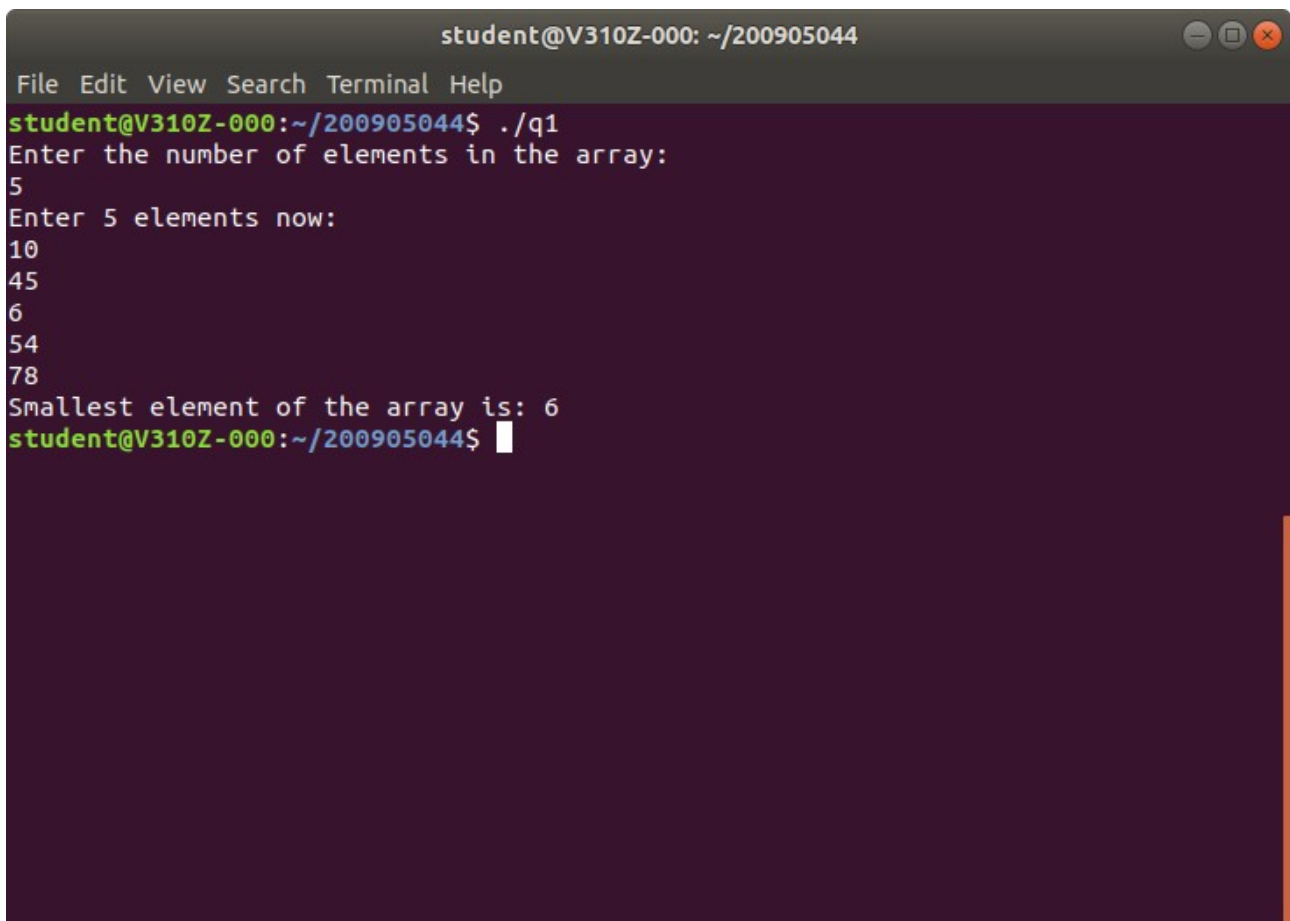
```

for (i = 0; i < n; ++i)
{
    scanf ("%d", a + i);
}

printf ("Smallest element of the array is: %d\n", Smallest (a, n));
return 0;
}

```

Sample Input and Output:



The screenshot shows a terminal window with the title "student@V310Z-000: ~/200905044". The terminal displays the following interaction:

```

student@V310Z-000:~/200905044$ ./q1
Enter the number of elements in the array:
5
Enter 5 elements now:
10
45
6
54
78
Smallest element of the array is: 6
student@V310Z-000:~/200905044$

```

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.

Program:

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

void
Multiply (int m1, int n1, int m2, int n2, int **a, int **b)
{
    int i, j, k, c[100][100];

    if (n1 != m2)

        {

            printf ("\nError Matrices cannot be Multiplied!\n");

            exit (0);

        }

    for (i = 0; i < m1; i++)

        {

            for (j = 0; j < n2; j++)

                {

                    c[i][j] = 0;

                    for (k = 0; k < n1; k++)

                        c[i][j] = c[i][j] + (a[i][k] * b[k][j]);

                }

        }

}
```

```

printf ("The Resultant Matrix after the Multiplication is :\n");

for (i = 0; i < m1; i++)

{
    for (j = 0; j < n2; j++)

        printf ("%d ", c[i][j]);

    printf ("\n");
}

}

```

```

int
main ()
{
    int m1, n1, m2, n2, **a, **b, i, j;
    printf ("Enter the number of rows in 1st matrix: \n");
    scanf ("%d", &m1);

    printf ("Enter the number of columns in 1st matrix: \n");
    scanf ("%d", &n1);

    printf ("Enter the number of rows in 2nd matrix: \n");
    scanf ("%d", &m2);

    printf ("Enter the number of columns in 2nd matrix: \n");
    scanf ("%d", &n2);

    a = (int **) calloc (m1 + 1, sizeof (int *));
    b = (int **) calloc (m2 + 1, sizeof (int *));

```

```
printf ("Enter elements of the 1st matrix:\n");
for (i = 0; i < m1; i++)
{
    a[i] = (int *) calloc (n1 + 1, sizeof (int));

    for (j = 0; j < n1; j++)
    {
        scanf ("%d", &a[i][j]);
    }
}
```

```
printf ("Enter elements of the 2nd matrix:\n");

for (i = 0; i < m2; i++)
{
    b[i] = (int *) calloc ((n2 + 1), sizeof (int));

    for (j = 0; j < n2; j++)
    {
        scanf ("%d", &b[i][j]);
    }
}
```

```
printf ("\nThe First Matrix is :\n");

for (i = 0; i < m1; i++)
{
    for (j = 0; j < n1; j++)

        printf ("%d ", a[i][j]);

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

```
printf ("\nThe Second Matrix is :\n");
```

```
for (i = 0; i < m2; i++)  
{  
    for (j = 0; j < n2; j++)  
        printf ("%d ", b[i][j]);  
    printf ("\n");  
}
```

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

```
Multiply (m1, n1, m2, n2, a, b);
```

```
for (i = 0; i < m1; i++)
```

```
    free (a[i]);
```

```
free (a);
```

```
for (i = 0; i < m2; i++)
```

```
    free (b[i]);
```

```
free (b);
```

```
return 0;
```

```
}
```

Sample Input and Output:

```
student@V310Z-000: ~/200905044
File Edit View Search Terminal Help
student@V310Z-000:~/200905044$ ./q2
Enter the number of rows in 1st matrix:
3
Enter the number of columns in 1st matrix:
2
Enter the number of rows in 2nd matrix:
2
Enter the number of columns in 2nd matrix:
4
Enter elements of the 1st matrix:
7 8 2 4 6 5
Enter elements of the 2nd matrix:
3 4 9 6 1 5 4 9

The First Matrix is :
7 8
2 4
6 5

The Second Matrix is :
3 4 9 6
1 5 4 9

The Resultant Matrix after the Multiplication is :
29 68 95 114
10 28 34 48
23 49 74 81
student@V310Z-000:~/200905044$
```

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.

Program:

```
#include<stdio.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;  
};
```

int main()

```
{  
    struct EMPLOYEE emp[10];  
    struct EMPLOYEE *ptr=emp;
```



```
int n,i=0;

printf("Enter the number of employees: ");

scanf("%d",&n);


while(i<n)

{  printf("Enter Name: ");

    scanf( "%s",&(ptr+i)->name );


    printf("Enter DOB(day,month,year): ");

    scanf("%d",&((ptr+i)->dob.day));

    scanf("%d",&((ptr+i)->dob.month));

    scanf("%d",&((ptr+i)->dob.year));


    printf("Enter House no: ");

    scanf("%d",&((ptr+i)->address.house_no));


    printf("Enter Zipcode: ");

    scanf("%d",&((ptr+i)->address.zipcode));


    printf("Enter State:\t");

    scanf("%s",&((ptr+i)->address.state));
```

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

```
i++;
```

```
}
```

```
i=0;
```

```
while(i<n)
```

```
{
```

```
printf("Name: %s\n",&((ptr+i)->name));
```

```
printf("DOB: %d-%d-%d\n",(ptr+i)->dob.day,(ptr+i)->dob.month,  
(ptr+i)->dob.year);
```

```
printf("House no: %d\n",(ptr+i)->address.house_no);
```

```
printf("Zipcode: %d\n",(ptr+i)->address.zipcode);
```

```
printf("State: %s\n\n",&((ptr+i)->address.state));
```

```
i++;
```

```
}
```

```
return 0;
```

```
}
```

Sample Input and Output:

```
student@V310Z-000: ~/200905044/lab1
File Edit View Search Terminal Help

student@V310Z-000:~/200905044/lab1$ ./q3
Enter the number of employees: 2
Enter Name: Praveen
Enter DOB(day,month,year): 31 03 2003
Enter House no: 406
Enter Zipcode: 500076
Enter State:   Telangana

Enter Name: Rakesh
Enter DOB(day,month,year): 27 02 2002
Enter House no: 402
Enter Zipcode: 500074
Enter State:   UP

Name: Praveen
DOB: 31-3-2003
House no: 406
Zipcode: 500076
State: Telangana

Name: Rakesh
DOB: 27-2-2002
House no: 402
Zipcode: 500074
State: UP

student@V310Z-000:~/200905044/lab1$
```

THANK YOU!