

# CSE ASSIGNMENT - 9

---

## Problem 1

---

### Question:

Write a C program to use pointer array and find the positive values and return more than one element to the main program.

### Code:

```
// Program to return an array from a function making use of pointers

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

int *arr_return(const int *inarr, int size)
{
    int *outarr, counter = 1;

    outarr = (int *)calloc(1, sizeof(int));
    *outarr = 1;

    for (int i = 0; i < size; i++)
    {
        if (*(inarr + i) > 0)
        {
            counter += 1;
            outarr = (int *)realloc(outarr, counter);
            *outarr = counter;
            *(outarr + counter - 1) = *(inarr + i);
        }
    }

    return outarr;
}

int main(int argc, char const *argv[])
{
    int n;

    printf("How many elements would you like : ");
    scanf(" %d", &n);

    int elem_list[n], *returning_arr;

    printf("Enter the elements one by one : ");

    for (int i = 0; i < n; i++)
```

```

{
    scanf(" %d", elem_list + i);
}

returning_arr = arr_return(elem_list, n);

for (int i = 1; i < *returning_arr; i++)
{
    printf("%d ", *(returning_arr + i));
}

printf("\n");
free(returning_arr);

return 0;
}

```

### Output:

```

How many elements would you like : 7
Enter the elements one by one : 1
-0
7
8
-4
-6
-2
1 7 8

```

## Problem 2

### Question:

*Write a C program to find number of students passed in their online certificate course. Consider the following constraints: Get student name, regno, course name and marks for given course. Check the student mark status, if the student got above 40 marks and consider as pass grade, otherwise assign as fail grade and ask them to reappear the exam, in order to complete the course.*

### Code:

```

// Program to grade a student

#include <stdio.h>

int main(int argc, char const *argv[])
{
    struct stud
    {
        char name[100], course[100];
    }
}

```

```

    int regno, mark;
} * some_student, some_stud;

some_student = &some_stud;

printf("Enter Name, Regno, Course and Marks in order : ");
scanf(" %s", some_student->name);
scanf(" %d", &some_student->regno);
scanf(" %s", some_student->course);
scanf(" %d", &some_student->mark);

if (some_student->mark > 40)
{
    printf("You have passed\n");
}
else
{
    printf("Please retake the test\n");
}

return 0;
}

```

### Output:

```

Enter Name, Regno, Course and Marks in order : BC
125
Mecha
70
You have passed

```

```

Enter Name, Regno, Course and Marks in order : BC
12
Mecha
32
Please retake the test

```

---

## Problem 3

---

### Question:

*Write a C program to implement dynamic memory allocation for matrix multiplication.*

### Code:

```

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

int main()

```

```

{
    int i, j, k, rf, rs, cf, cs;
    printf("Enter the rows and columns of the first matrix: ");
    scanf("%d %d", &rf, &cf);
    printf("\nEnter the rows and columns of the second matrix: ");
    scanf("%d %d", &rs, &cs);
    while (cf != rs)
    {
        printf("\nInvalid matrix, try again: \n");
        printf("\nEnter the rows and columns of the second matrix: ");
        scanf("%d %d", &rs, &cs);
    }
    int mat1[rf][cf], mat2[rs][cs], *prod;
    printf("\nEnter the elements for matrix 1: ");
    for (i = 0; i < rf; i++)
    {
        for (j = 0; j < cf; j++)
        {
            scanf("%d", &mat1[i][j]);
        }
    }
    printf("\nEnter the elements for matrix 2: ");
    for (i = 0; i < rs; i++)
    {
        for (j = 0; j < cs; j++)
        {
            scanf("%d", &mat2[i][j]);
        }
    }
    prod = (int *)calloc(rf * cs, sizeof(int));
    for (i = 0; i < rf; i++)
    {
        for (j = 0; j < cs; j++)
        {
            for (k = 0; k < cf; k++)
            {
                prod[i + j] += mat1[i][k] * mat2[k][j];
            }
        }
    }
    for (i = 0; i < rf; i++)
    {
        for (j = 0; j < cs; j++)
        {
            printf(" %d ", prod[i + j]);
        }
        printf("\n");
    }
    free(prod);
}

```

**Output:**

Enter the rows and columns of the first matrix: 3  
3

Enter the rows and columns of the second matrix: 3  
3

Enter the elements for matrix 1: 1

2  
3  
4  
1  
2  
3  
4  
1

Enter the elements for matrix 2: 1

2  
3  
4  
1  
2  
3  
4  
1

18 30 49  
30 49 30  
49 30 18