



CP ❤
solutions !

with a lot of love

ft. Ayush's Version



```
1 // Ayush Codes...
2 // Write a program to generate all the three digits and four digits armstrong numbers
3
4 #include<stdio.h>
5 #include<math.h>
6
7
8 int noofdigits(int n){
9     int result=0;
10    while(n>0){
11        n/=10;
12        result++;
13    }
14    return result;
15 }
16
17 int main(){
18     for(int i=100; i<=9999; i++){
19         int t=i,sum=0;
20         while(t>0){
21             sum+=pow(t%10,noofdigits(i));
22             t/=10;
23         }
24         if (sum==i) printf("%d\n", i);
25     }
26 }
```



```
1 // Ayush Codes...
2 // Write a program to find a five digit number which on multiplication by 4 reverses its order
3
4 #include <stdio.h>
5
6 int reverse(int n){
7     int result=0;
8     while(n>0){
9         result=result*10+n%10;
10        n/=10;
11    }
12    return result;
13 }
14
15 int main() {
16     for(int i=10000; i<=25000; i++){
17         if (i*4==reverse(i)) printf("The number is : %d\n", i);
18     }
19 }
20
```



```
1 // Ayush Codes...
2 // Write a program to generate a given number triangle
3 // 1
4 // 1 2
5 // 1 2 3
6 // 1 2 3 4.
7 // .. N Terms
8
9
10 #include<stdio.h>
11
12 int main() {
13     int n; scanf("%d",&n);
14     for(int i = 1; i <= n; i++) {
15         for(int j = 1; j <= i; j++) {
16             printf("%d ", j);
17         }
18         printf("\n");
19     }
20 }
```



```
1 // Ayush Codes...
2 // Write a program to generate a given number triangle
3 // 1
4 // 2 3
5 // 3 4 5
6 // 4 5 6 7
7 // 5 6 7 8 9
8 // 6 7 8 9 0 1
9 // 7 8 9 0 1 2 3
10 // ... N Terms
11
12 #include<stdio.h>
13
14 int main() {
15     int n; scanf("%d", &n);
16     for(int i = 1; i <= n; i++) {
17         for(int j = i; j < i + i; j++) {
18             printf("%d ", j%10);
19         }
20         printf("\n");
21     }
22 }
23
```



```
1 // Ayush Codes...
2 // Write a program to print following number triangle
3 //           1           1
4 //           1 2 1       4
5 //           1 2 3 2 1   9
6 //           1 2 3 4 3 2 1   16
7 // .....N lines
8
9 #include <stdio.h>
10 int main() {
11     int n; scanf("%d", &n);
12
13     for (int i = 1; i <= n; i++) {
14         for (int j = i; j < n; j++) printf(" ");
15         for (int j = 1; j <= i; j++) printf("%d ", j);
16         for (int j = i - 1; j >= 1; j--) printf("%d ", j);
17         for (int j = n; j >= i; j--) printf(" ");
18         printf("%d", i*i);
19         printf("\n");
20     }
21
22 }
```



```
1 // Ayush Codes...
2 // Jee me padha hoga nCr = n-1Cr-1 + n-1Cr
3 // Write a program to evaluate the NCR Number using Functions where NCR=N!/R!(N-R)!
4
5 #include<stdio.h>
6
7 int ncr(int n, int r){
8     if(r==0 || r==n) return 1;
9     return ncr(n-1,r-1)+ncr(n-1,r);
10 }
11
12 int main(){
13     printf("Enter the value of n: ");
14     int n; scanf("%d",&n);
15     printf("Enter the value of r: ");
16     int r; scanf("%d",&r);
17     printf("The value of nCr is: %d\n",ncr(n,r));
18 }
```



```
1 // Write a program for printing following number Series S = X - X^3/3! + X^5/5! -  
2 // ....Term>=10^-5  
3 // This is expansion of sin(x)  
4  
5 #include <stdio.h>  
6 #include <math.h>  
7  
8 double calculate_sin(double x) {  
9     x = fmod(x + 3.141, 2 * 3.141) - 3.141; // because sin(x) = sin(x + 2 * pi)  
10    double term = x;  
11    double sum = term;  
12    int n = 1;  
13    while (fabs(term) > 1e-5) {  
14        term *= -x * x / ((2 * n) * (2 * n + 1));  
15        sum += term;  
16        n++;  
17    }  
18    return sum;  
19}  
20  
21 int main() {  
22     double x;  
23     printf("Enter the value of x: ");  
24     scanf("%lf", &x);  
25     printf("Sum of series = %lf\n", calculate_sin(x));  
26  
27     return 0;  
28 }
```



```
1 // Write a program to print the following summation series S=100 + 1 - X^2/2! + X^4/4! - X^5/5!
2 // + ....Term>10^-5
3 // 100 ke aage wali series is of cos(x)
4 #include <stdio.h>
5 #include <math.h>
6
7 double calculate_cos(double x) {
8     x = fmod(x + 3.141, 2 * 3.141) - 3.141; // because cos(x) = cos(x + 2 * pi)
9     double term = 1.0;
10    double sum = term;
11    int n = 1;
12    while (fabs(term) > 1e-5) {
13        term *= -x * x / (2 * n * (2 * n - 1));
14        sum += term;
15        n++;
16    }
17    return sum;
18 }
19
20 int main() {
21     double x;
22     printf("Enter the value of x: ");
23     scanf("%lf", &x);
24
25     double result = 100 + calculate_cos(x);
26     printf("Sum of series = %.5lf\n", result);
27
28     return 0;
29 }
```



```
1 // Write a program to print the following summation series S=(a+b)/(c+d)/1! + (a+2b)/(c+d)^2/2!
2 // +(a+3b)/(c+d)^3/3!..3,5,9 Term
3
4 #include <stdio.h>
5 #include <math.h>
6
7 long fact(int n){
8     if(n==0 || n==1) return 1;
9     return n * fact(n-1);
10 }
11 double calcsum(double a, double b, double c, double d, int n){
12     double sum = 0;
13     for(int i = 1; i <= n; i++){
14         sum += (a+i*b) / pow(c+d,i)*fact(i)*1.0;
15     }
16     return sum;
17 }
18
19 int main(){
20     double a,b,c,d;
21     scanf("%lf %lf %lf %lf", &a, &b, &c, &d);
22     printf("Sum upto 3 terms : %lf\n", calcsum(a,b,c,d,3));
23     printf("Sum upto 5 terms : %lf\n", calcsum(a,b,c,d,5));
24     printf("Sum upto 9 terms : %lf\n", calcsum(a,b,c,d,9));
25 }
```



```
1 // Write a program to print the following summation series S= (e^x+1)/(log x + p) +
2 // (e^x+1)^2/(log x + 2p) + (e^x+1)^3/(log x + 3p) + ..... + 10 terms
3
4 #include <stdio.h>
5 #include <math.h>
6
7 double sum(double x, double p, int n) {
8     double sum = 0;
9     for (int i = 1; i <= n; i++) {
10         sum += pow(exp(x)+1,i)/(log(x)+i*p);
11     }
12     return sum;
13 }
14
15 int main() {
16     double x, p;
17     printf("Enter the value of x and p: ");
18     scanf("%lf %lf", &x, &p);
19     printf("Sum of series upto 10 terms = %lf\n", sum(x, p, 10));
20
21     return 0;
22 }
```



```
1 // Program to find the sum of the digits of the number and reverse of the number
2
3 #include<stdio.h>
4 int reverse(int n){
5     int rev = 0;
6     while(n){
7         rev = rev*10 + n%10;
8         n /= 10;
9     }
10    return rev;
11 }
12 int sumofdigits(int n){
13     int sum = 0;
14     while(n){
15         sum += n%10;
16         n /= 10;
17     }
18    return sum;
19 }
20
21 int main(){
22     int n; scanf("%d", &n);
23     printf("Reverse : %d\nSum of digits: %d\n", reverse(n), sumofdigits(n));
24 }
```



```
1 // Write a program to find the area of a Triangle using Functions Area=SQRT(S*(S-a)*(S-b)*(S-c))
2 // where S=(a+b+c)/2.0
3
4 #include <stdio.h>
5 #include <math.h>
6
7 int isvalid(float a, float b, float c){
8     if(a+b>c && b+c>a && c+a>b) return 1;
9     return 0;
10 }
11
12 double calculate_area(float a, float b, float c){
13     if(isvalid(a,b,c)==1){
14         float s = (a+b+c)/2.0;
15         return sqrt(s*(s-a)*(s-b)*(s-c));
16     }
17     else return -1;
18 }
19 int main(){
20     float a,b,c;
21     printf("Enter the sides of the triangle: ");
22     scanf("%f %f %f", &a, &b, &c);
23     printf("Area of the triangle: %lf\n",
24           (calculate_area(a,b,c)!=0) ? calculate_area(a,b,c) : printf("Invalid triangle\n"));
25 }
26
```

```

1 // Write a program using functions to verify
2 // ([A] * [B])T = [B]T * [A]T
3
4 #include<stdio.h>
5
6 void printMatrix(int arr[100][100], int m){
7     for(int i=0; i<m; i++){
8         for(int j=0; j<m; j++){
9             printf("%d ", arr[i][j]);
10        }
11    }
12    printf("\n");
13}
14}
15
16
17 void trans(int x[100][100], int y[100][100], int m){
18    for(int i=0; i<m; i++){
19        for(int j=0; j<m; j++){
20            y[j][i] = x[i][j];
21        }
22    }
23}
24}
25
26
27
28 void multiplyMatrix(int a[100][100], int b[100][100],int c[100][100], int m){
29     //Initialize matrix result with zero
30     for(int i=0; i<m; i++){
31         for(int j=0; j<m; j++){
32             c[i][j] = 0;
33         }
34     }
35
36     //Calculate Matrix Result
37     for(int i=0; i<m; i++){
38         for(int j=0; j<m; j++){
39             for(int k=0; k<m; k++){
40                 c[i][j] += a[i][k]*b[k][j];
41             }
42         }
43     }
44 }
45
46 int main(){
47     int m, t;
48     printf("Enter order of matrix A and B: ");
49     scanf("%d",&m);
50
51     int A[100][100];
52     int At[100][100];
53     int B[100][100];
54     int Bt[100][100];
55     int mul1[100][100];
56     int mul2[100][100];
57
58     printf("Enter A:\n");
59     for(int i=0; i<m; i++){
60         for(int j=0; j<m; j++){
61             scanf("%d", &A[i][j]);
62         }
63     }
64     trans(A,At,m);
65
66     printf("Enter B:\n");
67     for(int i=0; i<m; i++){
68         for(int j=0; j<m; j++){
69             scanf("%d", &B[i][j]);
70         }
71     }
72     trans(B,Bt,m);
73
74     multiplyMatrix(A,B,mul1,m);
75     multiplyMatrix(Bt, At, mul2, m);
76
77     int mul1Trans[100][100];
78     trans(mul1, mul1Trans, m);
79     trans(mul1, mul1Trans, m);
80
81     printf("(A*B)^T\n");
82     printMatrix(mul1Trans,m);
83     printf("(B^T)*(A^T)\n");
84     printMatrix(mul2, m);
85
86     int flag=0;
87     for(int i=0; i<m; i++){
88         for(int j=0; j<m; j++){
89             if(mul1Trans[i][j] == mul2[i][j]){
90                 flag = 0;
91             }
92             else {flag=1; printf("Not Verified"); break;}
93         }
94     }
95
96     if(flag==0) printf("Verified\n");
97 }
98

```



```
1 // Write a program to generate all the prime numbers between 300 to 500, print 5 number per line
2
3 #include <stdio.h>
4
5 int is_prime(int num) {
6     if (num <= 1) return 0;
7     for (int i = 2; i * i <= num; i++) if (num % i == 0) return 0;
8     return 1;
9 }
10
11 int main() {
12     int count = 0;
13     for (int i = 300; i <= 500; i++) {
14         if (is_prime(i)==1) {
15             printf("%d ", i);
16             count++;
17             if (count % 5 == 0) printf("\n");
18         }
19     }
20 }
```



```
1 // Write a program to generate all Pythagorean triplets within 100
2
3 #include <stdio.h>
4
5 int main() {
6     int i, j, k;
7     for (i = 1; i <= 100; i++) {
8         for (j = i + 1; j <= 100; j++) {
9             for (k = j + 1; k <= 100; k++) {
10                if (i * i + j * j == k * k) {
11                    printf("%d %d %d\n", i, j, k);
12                }
13            }
14        }
15    }
16 }
```



```
1 // Write a Program to find the biggest and smallest of an array of integers
2
3 #include<stdio.h>
4
5 int main(){
6     int n; printf("Size of arr: "); scanf("%d",&n);
7     int arr[n];
8     for(int i=0; i<n; i++) scanf("%d", &arr[i]);
9     int big=arr[0], small=arr[0];
10    for(int i=1; i<n; i++) if(arr[i]>big) big=arr[i];
11    for(int i=1; i<n; i++) if(arr[i]<small) small=arr[i];
12
13    printf("Biggest: %d\n", big);
14    printf("Smallest: %d\n", small);
15 }
```



```
1 // Write Program to find the biggest and smallest element and interchange them in the initialized
2 // array
3
4 #include<stdio.h>
5
6 void swap(int arr[],int i, int j){
7     int temp = arr[i];
8     arr[i] = arr[j];
9     arr[j] = temp;
10 }
11
12 int main(){
13     int n; printf("Size of arr: "); scanf("%d",&n);
14     int arr[n];
15     for(int i=0; i<n; i++) scanf("%d",&arr[i]);
16     int bigIndex=0, smallIndex=0;
17     for(int i=1; i<n; i++) if(arr[i]>arr[bigIndex]) bigIndex=i;
18     for(int i=1; i<n; i++) if(arr[i]<arr[smallIndex]) smallIndex=i;
19     swap(arr, bigIndex, smallIndex);
20     printf("Array after swapping biggest and smallest : \n");
21     for(int i=0; i<n; i++) printf("%d ",arr[i]);
22 }
```



```
1 // Write a program to sort an array in ascending/descending order using Selection Sort
2
3 #include <stdio.h>
4
5 void selectionSort(int arr[], int n, int choice) {
6     int i, j, min_idx, max_idx, temp;
7
8     for (i = 0; i < n-1; i++) {
9         if (choice==1) {
10             min_idx = i;
11             for (j = i+1; j < n; j++)
12                 if (arr[j] < arr[min_idx])
13                     min_idx = j;
14             temp = arr[min_idx];
15             arr[min_idx] = arr[i];
16             arr[i] = temp;
17         } else {
18             max_idx = i;
19             for (j = i+1; j < n; j++)
20                 if (arr[j] > arr[max_idx])
21                     max_idx = j;
22             temp = arr[max_idx];
23             arr[max_idx] = arr[i];
24             arr[i] = temp;
25         }
26     }
27 }
28
29 int main() {
30     int n, i, choice;
31     printf("Enter number of elements: "); scanf("%d", &n);
32     int arr[n];
33     printf("Enter %d elements: ", n);
34     for (i = 0; i < n; i++) scanf("%d", &arr[i]);
35
36     printf("Choose sorting order:\n1. Ascending\n2. Descending\nEnter your choice: ");
37     scanf("%d", &choice);
38
39     selectionSort(arr, n, choice);
40
41     printf("Sorted array: ");
42     for (i = 0; i < n; i++) printf("%d ", arr[i]);
43     printf("\n");
44 }
```



```
1 // Write a program to sort an array in ascending/descending order using Bubble Sort
2
3 #include <stdio.h>
4
5 void bubbleSort(int arr[], int n, int choice) {
6     int i, j, temp;
7     for (i = 0; i < n-1; i++) {
8         for (j = 0; j < n-i-1; j++) {
9             if ((choice==1 && arr[j] > arr[j+1]) || (choice!=1 && arr[j] < arr[j+1])) {
10                 temp = arr[j];
11                 arr[j] = arr[j+1];
12                 arr[j+1] = temp;
13             }
14         }
15     }
16 }
17
18
19 int main() {
20     int n, i, choice;
21     printf("Enter number of elements in the array: ");
22     scanf("%d", &n);
23     int arr[n];
24     printf("Enter the elements of the array:\n");
25     for (i = 0; i < n; i++) scanf("%d", &arr[i]);
26     printf("Choose sorting order:\n1. Ascending\n2. Descending\n");
27     scanf("%d", &choice);
28     bubbleSort(arr, n, choice);
29     printf("Sorted array:\n");
30     for (i = 0; i < n; i++) printf("%d ", arr[i]);
31 }
32
```



```
1 // Write a program to read N random numbers and evaluate the sum and average of the same
2
3 #include <stdio.h>
4
5 int main(){
6     int n; printf("Enter the number of elements: "); scanf("%d", &n);
7     int arr[n];
8     for(int i=0; i<n; i++) scanf("%d", &arr[i]);
9     int sum=0;
10    for(int i=0; i<n; i++) sum+=arr[i];
11    printf("Sum: %d\n", sum);
12    printf("Average: %f\n", (float)sum/n);
13 }
```



```
1 // It is observed that all integers converges to 1 is subjected to the following process
2 // a) The number is even divide by 2
3 // b) if it is odd multiply by three and add 1
4 // Write a Program to find all the intermediate values till it converges to 1 and also count number of
5 // iterations
6
7 #include <stdio.h>
8
9 int main(){
10     int n;
11     printf("Enter a number: ");
12     scanf("%d", &n);
13     int count = 0;
14     while(n != 1){
15         if(n % 2 == 0) n/=2;
16         else n=3*n+1;
17         printf("%d ", n);
18         count++;
19     }
20     printf("\n");
21     printf("Number of iterations: %d\n", count);
22 }
```



```
1 // Write a program to verify whether the given number is odd, a perfect square, an Armstrong
2 // number, a term in the Fibonacci series or it is a prime number
3
4 #include <stdio.h>
5 #include <math.h>
6
7 void isOdd(int n){
8     if(n%2!=0) printf("%d is an odd number\n", n);
9     else printf("%d is not an odd number\n", n);
10 }
11
12 void isPerfectSquare(int n){
13     if(ceil(sqrt(n))==floor(sqrt(n))) printf("%d is a perfect square\n", n);
14     else printf("%d is not a perfect square\n", n);
15 }
16
17 int noofdigits(int n){
18     int count = 0;
19     while(n>0){
20         count++;
21         n/=10;
22     }
23     return count;
24 }
25
26 void isArmstrong(int n){
27     int sum = 0;
28     int temp = n;
29     while(temp>0){
30         int rem = temp%10;
31         sum += pow(rem,noofdigits(n));
32         temp/=10;
33     }
34     if(sum==n) printf("%d is an Armstrong number\n", n);
35     else printf("%d is not an Armstrong number\n", n);
36 }
37
38 void is_fib(int n){
39     int a = 0, b = 1, c;
40     while(a<=n){
41         if(a==n){
42             printf("%d is a term in the Fibonacci series\n", n);
43             return;
44         }
45         c = a+b;    a = b;    b = c;
46     }
47     printf("%d is not a term in the Fibonacci series\n", n);
48 }
49
50 void isPrime(int n){
51     int flag = 0;
52     for(int i=2; i<=n/2; i++){
53         if(n%i==0){
54             flag = 1;
55             break;
56         }
57     }
58     if(flag==0) printf("%d is a prime number\n", n);
59     else printf("%d is not a prime number\n", n);
60 }
61
62 int main(){
63     int n;
64     printf("Enter a number: ");
65     scanf("%d", &n);
66     isOdd(n);
67     isPerfectSquare(n);
68     isArmstrong(n);
69     is_fib(n);
70     isPrime(n);
71 }
```



```
1 #include <stdio.h>
2 #include <string.h>
3
4 struct st {
5     char name[100];
6     int id, p, c, m, total;
7 };
8
9 void swap(struct st *a, struct st *b) {
10     struct st temp = *a;
11     *a = *b;
12     *b = temp;
13 }
14
15 int main() {
16     int n;
17     printf("Enter the number of students: ");
18     scanf("%d", &n);
19     struct st st[n];
20     for (int i = 0; i < n; i++) {
21         printf("Enter the name, id, and marks of P, C, M of student %d: ", i + 1);
22         scanf("%s %d %d %d", st[i].name, &st[i].id, &st[i].p, &st[i].c, &st[i].m);
23         st[i].total = st[i].p + st[i].c + st[i].m;
24     }
25     for (int i = 0; i < n - 1; i++) {
26         for (int j = 0; j < n - i - 1; j++) {
27             if (st[j].total < st[j + 1].total ||
28                 (st[j].total == st[j + 1].total && st[j].m < st[j + 1].m) ||
29                 (st[j].total == st[j + 1].total && st[j].m == st[j + 1].m && st[j].p < st[j + 1].p) ||
30                 (st[j].total == st[j + 1].total && st[j].m == st[j + 1].m && st[j].p == st[j + 1].p && st[j].c < st[j + 1].c)) {
31                 swap(&st[j], &st[j + 1]);
32             }
33         }
34     }
35     printf("Merit List:\n");
36     for (int i = 0; i < n; i++) {
37         printf("Name: %s\nID: %d\nPhysics: %d\nChemistry: %d\nMaths: %d\nTotal: %d\n\n", st[i].name, st[i].id, st[i].p, st[i].c, st[i].m, st[i].total);
38     }
39 }
40 }
```



```
1 // Write a program to store the details of the students using structures sort student details in
2 // alphabetical order
3
4 // They haven't mentioned what details so I'll take it as name
5
6 #include <stdio.h>
7 #include <string.h>
8
9 struct st {
10     char name[100];
11 };
12
13 void swap(struct st *a, struct st *b) {
14     struct st temp = *a;
15     *a = *b;
16     *b = temp;
17 }
18
19 int main() {
20     int n;
21     printf("Enter the number of students: ");
22     scanf("%d", &n);
23     struct st st[n];
24     for (int i = 0; i < n; i++) {
25         printf("Enter the name of student %d: ", i + 1);
26         scanf("%s", st[i].name);
27     }
28     for (int i = 0; i < n - 1; i++) {
29         for (int j = 0; j < n - i - 1; j++) {
30             if (strcmp(st[j].name, st[j + 1].name) > 0) {
31                 swap(&st[j], &st[j + 1]);
32             }
33         }
34     }
35     printf("Sorted List:\n");
36     for (int i = 0; i < n; i++) {
37         printf("Name: %s\n", st[i].name);
38     }
39 }
```



```
1 // Write a program to read details of students like Roll No Name and Class, arrange the details in
2 // alphabetic order of names using arrays
3
4 #include <stdio.h>
5 #include <string.h>
6
7 struct st {
8     char name[100];
9     int roll;
10    char class[100];
11 };
12
13 void swap(struct st *a, struct st *b) {
14     struct st temp = *a;
15     *a = *b;
16     *b = temp;
17 }
18
19 int main() {
20     int n;
21     printf("Enter the number of students: ");
22     scanf("%d", &n);
23     struct st st[n];
24     for (int i = 0; i < n; i++) {
25         printf("Enter the name, roll number, and class of student %d: ", i + 1);
26         scanf("%s %d %s", st[i].name, &st[i].roll, st[i].class);
27     }
28     for (int i = 0; i < n - 1; i++) {
29         for (int j = 0; j < n - i - 1; j++) {
30             if (strcmp(st[j].name, st[j + 1].name) > 0) {
31                 swap(&st[j], &st[j + 1]);
32             }
33         }
34     }
35     printf("Sorted List:\n");
36     for (int i = 0; i < n; i++) {
37         printf("Name: %s\nRoll Number: %d\nClass: %s\n", st[i].name, st[i].roll, st[i].class);
38     }
39 }
40
```

```

1 // With the help of methods for inserting the elements in an sorted array, method for deleting an
2 // element from array,
3 // method for sorting an array, method for merging two arrays,
4 // method for searching in an array. Write a menu driven program
5 // (Switch Case) to repeatedly execute the one or more
6 // of this methods
7
8 #include <stdio.h>
9
10 void insert(int arr[], int *n, int ele) {
11     int i = *n - 1;
12     while (i >= 0 && arr[i] > ele) arr[i + 1] = arr[i--];
13     arr[i + 1] = ele;
14     (*n)++;
15 }
16
17 void delete(int arr[], int *n, int ele) {
18     int i = 0;
19     while (i < *n && arr[i] != ele) i++;
20     if (i == *n) return;
21     while (i < *n - 1) arr[i] = arr[i + 1], i++;
22     (*n]--;
23 }
24
25 void sort(int arr[], int n) {
26     for (int i = 0; i < n - 1; i++) {
27         for (int j = 0; j < n - i - 1; j++) {
28             if (arr[j] > arr[j + 1]) {
29                 int temp = arr[j];
30                 arr[j] = arr[j + 1];
31                 arr[j + 1] = temp;
32             }
33     }
34 }
35 void merge(int arr1[], int n1, int arr2[], int n2, int arr3[], int *n3) {
36     *n3 = n1 + n2;
37     for (int i = 0; i < n1; i++) arr3[i] = arr1[i];
38     for (int i = 0; i < n2; i++) arr3[n1 + i] = arr2[i];
39     sort(arr3, *n3);
40 }
41
42 int search(int arr[], int n, int ele) {
43     for (int i = 0; i < n; i++)
44         if (arr[i] == ele) return i;
45     return -1;
46 }
47
48 void printArray(int arr[], int n) {
49     for (int i = 0; i < n; i++) printf("%d ", arr[i]);
50     printf("\n");
51 }
52
53 int main() {
54     int n, choice, ele, n2, n3;
55     printf("Enter the number of elements in the array: ");
56     scanf("%d", &n);
57     int arr[n], arr2[100], arr3[200];
58     printf("Enter the elements of the array: ");
59     for (int i = 0; i < n; i++) scanf("%d", &arr[i]);
60
61     do {
62         printf("1. Insert\n2. Delete\n3. Sort\n4. Merge\n5. Search\n6. Exit\nEnter your choice: ");
63         scanf("%d", &choice);
64         switch (choice) {
65             case 1:
66                 printf("Enter the element to insert: ");
67                 scanf("%d", &ele);
68                 insert(arr, &n, ele);
69                 printf("Array after insertion: ");
70                 printArray(arr, n);
71                 break;
72             case 2:
73                 printf("Enter the element to delete: ");
74                 scanf("%d", &ele);
75                 delete(arr, &n, ele);
76                 printf("Array after deletion: ");
77                 printArray(arr, n);
78                 break;
79             case 3:
80                 sort(arr, n);
81                 printf("Array after sorting: ");
82                 printArray(arr, n);
83                 break;
84             case 4:
85                 printf("Enter the number of elements in the second array: ");
86                 scanf("%d", &n2);
87                 printf("Enter the elements of the second array: ");
88                 for (int i = 0; i < n2; i++) scanf("%d", &arr2[i]);
89                 merge(arr, n, arr2, n2, arr3, &n3);
90                 printf("Merged array: ");
91                 printArray(arr3, n3);
92                 break;
93             case 5:
94                 printf("Enter the element to search: ");
95                 scanf("%d", &ele);
96                 int pos = search(arr, n, ele);
97                 if (pos == -1) printf("Element not found\n");
98                 else printf("Element found at position %d\n", pos + 1);
99                 break;
100            case 6:
101                break;
102            default:
103                printf("Invalid choice\n");
104        }
105    } while (choice != 6);
106 }
107

```



```
1 // Write a program using functions to evaluate
2 // [T]=[A]*[B] +[([C] * [D])2]T + [P]4
3 #include<stdio.h>
4
5 void printMatrix(int arr[100][100], int m){
6     for(int i=0; i<m; i++){
7         for(int j=0; j<m; j++) printf("%d ", arr[i][j]);
8         printf("\n");
9     }
10 }
11
12 void trans(int x[100][100], int y[100][100], int m){
13     for(int i=0; i<m; i++) for(int j=0; j<m; j++) y[j][i] = x[i][j];
14 }
15
16 void multiplyMatrix(int a[100][100], int b[100][100], int c[100][100], int m){
17     for(int i=0; i<m; i++) for(int j=0; j<m; j++) c[i][j] = 0;
18
19     for(int i=0; i<m; i++) for(int j=0; j<m; j++) for(int k=0; k<m; k++)
20         c[i][j] += a[i][k] * b[k][j];
21 }
22
23 void addMatrix(int a[100][100], int b[100][100], int c[100][100], int m){
24     for(int i=0; i<m; i++) for(int j=0; j<m; j++)
25         c[i][j] = a[i][j] + b[i][j];
26 }
27
28 int main(){
29     int A[100][100], B[100][100], C[100][100], D[100][100],
30         N[100][100], P[100][100], Q[100][100], T[100][100], Tr[100][100], Qt[100][100], res[100][100];
31     int m;
32
33     printf("Enter Order of matrices: ");
34     scanf("%d", &m);
35
36     printf("Enter A:\n");
37     for(int i=0; i<m; i++) for(int j=0; j<m; j++)
38         scanf("%d", &A[i][j]);
39
40     printf("Enter B:\n");
41     for(int i=0; i<m; i++) for(int j=0; j<m; j++)
42         scanf("%d", &B[i][j]);
43
44     printf("Enter C:\n");
45     for(int i=0; i<m; i++) for(int j=0; j<m; j++)
46         scanf("%d", &C[i][j]);
47
48     printf("Enter D:\n");
49     for(int i=0; i<m; i++) for(int j=0; j<m; j++)
50         scanf("%d", &D[i][j]);
51
52     printf("Enter P:\n");
53     for(int i=0; i<m; i++) for(int j=0; j<m; j++)
54         scanf("%d", &P[i][j]);
55
56     multiplyMatrix(A, B, N, m);
57     multiplyMatrix(C, D, Q, m);
58     multiplyMatrix(Q, Q, Q, m);
59     trans(Q, Qt, m);
60
61     multiplyMatrix(A, A, A, m);
62     multiplyMatrix(A, A, A, m);
63
64     addMatrix(N, Q, T, m);
65     addMatrix(T, Qt, Tr, m);
66     addMatrix(Tr, A, res, m);
67
68     multiplyMatrix(P, P, P, m);
69     multiplyMatrix(P, P, P, m);
70     addMatrix(res, P, res, m);
71
72     printf("Result T = \n");
73     printMatrix(res, m);
74 }
75 }
```