INDEX

1. To sort single dimensional array by bubble sorting method (ascending)
2. To add two dimensional array by using function
3. To multiply two dimensional array by using function
4. To read and print file by line.by use of malloc.
5. To print in file read odd lines and print in file named odd , and even lines in file named even
6. To find factorial using function
7. To find factorial using recursive function
8. To swap numbers using function (pass by reference )
9. To find fibonacci series using recursive function
10. To print pattern1
11. To print pattern2
12. To concatenate two strings without string function
13. To reverse the string and print it(without other array)
14. To read formated file input and store it in a structure and then print it
15. To take input and store it in structure and then print info of student with maximum marks.
16. To find Sub-sequence in the string
17. To find Sub-String in the string
18. To find maximum number out of 3 using ternary operator
19. Find Greatest common divisor 0f two positive integer
20. To Calculate the Average of set of N numbers

21) Reverse the given number

22) Find sum of nth. series (1/1! + 2/2! + 3/3! +..... n/n!)

23) To Calculate simple interest

24) Swap two numbers without using third variable

25) To take input and store it in structure and then print info of student with maximum marks by passing structure by reference

//Program details -To sort single dimensional array by bubble sorting method (ascending)

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<stdio.h>

void printarray(int x[],int size)

{

int i ;

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

{

printf("%d" , x[i]);

}

}

int main()

{

int i , j = 0, temp =0 ;

int a[10]= {5,3,8,9,2,1,4,7,6,0};

for (i = 0 ; i <10 ; i++){

for (j = 0 ; j < (10-i) ; j++){

if (a[j]>a[j+1])

{

temp = a[j] ;

a[j] = a[j+1] ;

a[j+1] = temp ;

}

}

}

printarray(a ,10 );

return 0 ;

}

//Program details -To add two dimensional array by using function

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<stdio.h>

void printarray(int\* x,int size1,int size2)

{

int i , j;

for (i=0 ;i < size1; i++){

for (j=0 ;j < size2 ; j++){

printf("%d \t" , \*x);

x++;

}

printf("\n");

}

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

}

int main()

{

int i , j = 0, temp =0 ;

int a[3][3]= {5,3,8,9,2,1,4,7,6};

int b[3][3]= {3,5,2,6,8,9,6,7,4};

int c[3][3] ;

for (i=0 ;i < 3; i++){

for (j=0 ;j < 3; j++){

c[i][j] = a[i][j] + b[i][j] ;

}

}

printarray(a,3,3);

printarray(b,3,3);

printarray(c,3,3);

return 0 ;

}

//Program details -To multiply two dimensional array by using function

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<stdio.h>

void printarray(int\* x,int size1,int size2)

{

int i , j;

for (i=0 ;i < size1; i++){

for (j=0 ;j < size2 ; j++){

printf("%d \t" , \*x);

x++;

}

printf("\n");

}

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

}

int main()

{

int i , j = 0,k =0 ;

int a[3][3]= {5,3,8,9,2,1,4,7,6};

int b[3][3]= {3,5,2,6,8,9,6,7,4};

int c[3][3] ;

for (i=0 ;i < 3; i++){

for (j=0 ;j < 3; j++){

c[i][j] = a[i][0] \* b[0][j] + a[i][1] \* b[1][j] +a[i][2] \* b[2][j] ;

}

}

printarray(a,3,3);

printarray(b,3,3);

printarray(c,3,3);

return 0 ;

}

//Program details -To read and print file by line. use malloc.

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

int main ()

{

// the file we want to read from

char filename[] = "test.txt";

// open the file for reading

FILE \*file = fopen(filename, "r");

// make sure the file opened properly

if(NULL == file)

{

fprintf(stderr, "Cannot open file: %s\n", filename);

return 1;

}

// set up the buffer to read the line into buffer.

size\_t buffer\_size = 250;

char \*buffer = malloc(buffer\_size \* sizeof(char));

// read each line and print it to the screen

int line\_number = 0;

while(-1 != getline(&buffer, &buffer\_size, file))

{

printf("%d: %s", ++line\_number, buffer);

}

fflush(stdout);

// make sure we close the file and clean up the buffer when we're

// finished

fclose(file);

free(buffer);

return 0;

}

//Program details -To print in file . read odd lines and print in file named odd , and even lines in file named even

#include <stdio.h>

#include<stdlib.h>

int main()

{

FILE \*fp1;FILE \*fp2;FILE \*fp3;

char s[800];

fp1= fopen ("test.txt","r");

fp2= fopen ("testodd.txt","w");

fp3=fopen ("testeven.txt","w");

int ans,x,y ,i;

if (fp1 != NULL)

{

int line\_number = 0;

while( fgets(s, 799, fp1) != NULL)

{

if (++line\_number % 2 == 0)

{

fputs(s,fp2);

}

else{

fputs(s,fp3);

}

// printf("%d: %s", ++line\_number, s);

}

}else printf("file not exist");

fflush(stdout);

// make sure we close the file and clean up the buffer when we're

// finished

fclose(fp1);fclose(fp2);fclose(fp3);

return 0;

}

//Program details -To find factorial using function

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<stdio.h>

int factorial(int);

int main() {

int f;

printf("enter number to find factorial");

scanf("%d",&f);

printf("%d" , factorial(f));

return (0);

}

int factorial(int a ){

int n = a ;

a=a-1;

for (; a> 1 ; a--)

{

n = n \* a ;

}

return n;

}

//Program details -To find factorial using recursive function

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<stdio.h>

int factorial(int);

int main() {

int f;

printf("enter number to find factorial");

scanf("%d",&f);

printf("%d" , factorial(f));

return (0);

}

int factorial(int n ){

if (n > 1)

return n \* factorial (n-1) ;

else

return 1;

}

//Program details -To swap numbers using function (pass by reference )

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<stdio.h>

void swap(int\*,int\*);

int main() {

int num1 , num2 ;

printf("enter number 1 ");

scanf("%d",&num1);

printf("enter number 2 ");

scanf("%d",&num2);

swap(&num1,&num2);

printf("swapping done \n");

printf("number 1 %d \nnumber 2 %d ",num1 ,num2);

return (0);

}

void swap(int\* a,int\* b ){

int temp ;

temp = \*a;

\*a = \*b ;

\*b = temp ;

}

//Program details -To find fibonacci series using recursive function

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<stdio.h>

int fibonacci(int);//declaration

int main()

{

int n, c;

printf("Enter number of terms in series \n");

scanf("%d",&n);

printf("Fibonacci series\n");

for ( c = 0 ; c <= (n-1) ; c++ ){

printf("%d\n", fibonacci(c));

}

return 0;

}

int fibonacci(int n)

{

if ( n == 0 )

return 0;

else if ( n == 1 )

return 1;

else

return ( fibonacci(n-1) + fibonacci(n-2) );

}

//Program details -To print following pattern1

/\*

1

212

32123

4321234

\*/

#include<stdio.h>

int main()

{

int i , j = 0,k =0, num = 0;

printf ("number of rows");

scanf("%d",&num);

for (i=0 ;i <= num; i++){

for (j=num;j > i; j--){

printf(" ");

}

for (j = i ; j > 1 ; j--){

printf("%d" , j );

}

for (j = 1 ; j <= i ; j++){

printf("%d" , j );

}

printf("\n");

}

return 0 ;

}

//Program details -To print following pattern

/\*

1

121

12321

1234321

12321

121

1

\*/

#include<stdio.h>

int main()

{

int i , j = 0,k =0 ,num = 0;

printf ("number of rows");

scanf("%d",&num);

for (i=0 ;i <= num; i++){

for (j=num;j > i; j--){

printf(" ");

}

for (j = 1 ; j < i ; j++){

printf("%d" , j );

}

for (j = i ; j >= 1 ; j--){

printf("%d" , j );

}

printf("\n");

}

for (i=(num-1) ;i > 0; i--){

for (j=0;j <=((num-1)-i); j++){

printf(" ");

}

for (j = 1 ; j < i ; j++){

printf("%d" , j );

}

for (j = i ; j >= 1 ; j--){

printf("%d" , j );

}

printf("\n");

}

return 0 ;

}

//Program details -To concatenate two strings without string function

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<stdio.h>

char strconcat(char s1[],char s2[]){

int len1,len2,j,i;

for(len1 = 0 ; s1[len1] != '\0';len1++);

for(len2 = 0 ; s2[len2] != '\0';len2++);

for ( i = len1, j = 0 ; i <len1+len2 ; i++ ,j++)

{

s1[i] = s2 [j];

}

s1[i] = '\0';

printf("concatenated string - %s",s1);

}

int main(){

char s1[20] ,s2[20];

printf("enter first string - ");

scanf("%s",s1);

printf("enter second string - ");

scanf("%s",s2);

strconcat( s1,s2);

return 0 ;

}

//Program details -To reverse the string and print it(without other array)

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<stdio.h>

int main()

{

char s[20],temp ;

int len,i;

printf("enter a string - ");

scanf("%s",s);

for(len = 0 ; s[len] != '\0';len++);

for (i = 0 ; i < (len/2) ;i++){

temp = s[i];

s[i] = s[(len-1)-i];

s[(len-1)-i] = temp ;

}

printf("reversed string is %s",s);

return 0 ;

}

//Program details -To read formated file input and store it in a structure and then print it

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<stdio.h>

struct student{

int rollno;

char name[40];

} stu[10] ;

int main()

{

FILE \*fp;

char s[800];

fp = fopen ("stu\_data.txt","r");

int i = 0, j = 0;

while (fscanf(fp,"%d %s",&stu[i].rollno,&stu[i].name )>0){

i++;

}

for(j = 0 ; j < 10 ; j ++ ){

printf("%d " ,stu[j].rollno) ;

printf("%s \n" ,stu[j].name) ;

}

fclose(fp);

return 0 ;

}

//Program details -To take input and store it in structure and then print info of student with maximum marks.

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<stdio.h>

struct student{

int rollno;

char name[40];

int marks;

} stu[3] ;

int main()

{

int i = 0, j = 0;

int max\_index = 0;

int max\_marks = 0;

for(i = 0 ; i < 3 ; i++ ){

printf("Enter roll number,name ,marks of student %d " ,i+1) ;

scanf("%d %s %d" ,&stu[i].rollno,&stu[i].name,&stu[i].marks) ;

}

for(j = 0 ; j < 3 ; j ++ ){

if (stu[j].marks > max\_marks)

{

max\_marks = stu[j].marks ;

max\_index = j ;

}

}

printf("maximum scorer rollno - %d \n" ,stu[max\_index].rollno) ;

printf("maximum scorer name %s \n" ,stu[max\_index].name) ;

printf("maximum scorer marks %d " ,stu[max\_index].marks) ;

return 0 ;

}

//Program details: To find Sub-sequence in the string.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<stdio.h>

#include<string.h>

void isSubSequence(char str1[], char str2[])

{

int j = 0 ,i; // For index of str1 (or subsequence

int n = strlen(str1);

int m = strlen(str2);

/\* Traverse str2 and str1, and compare current character

of str2 with first unmatched char of str1, if matched

then move ahead in str1 \*/

for (i =0; i<n && j<m; i++)

{

if (str2[j] == str1[i]){

j++;

}

}

printf("%d",j);

// If all characters of str1 were found in str2

if (j == m)

printf("subsequence");

else

printf("not a subsequence");

}

int main()

{

char str1[20] ,str2[20] ;

printf("enter string 1");

scanf("%s",str1);

printf("enter string 2");

scanf("%s",str2);

isSubSequence(str1, str2);

return 0; }

//Program details: To find Sub-String in the string.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<stdio.h>

#include<string.h>

int search(char[], char[]);

int main() {

int loc;

char str1[20] ,str2[20] ;

printf("enter string 1");

scanf("%s",str1);

printf("enter string 2");

scanf("%s",str2);

loc = search(str1, str2);

if (loc == -1)

printf("\nNot found");

else

printf("\nFound at location %d", loc + 1);

return (0);

}

int search(char src[], char str[]) {

int i, j, firstOcc;

i = 0, j = 0;

while (src[i] != '\0') {

while (src[i] != str[0] && src[i] != '\0')

i++;

if (src[i] == '\0')

return (-1);

firstOcc = i;

while (src[i] == str[j] && src[i] != '\0' && str[j] != '\0') {

i++;

j++;

}

if (str[j] == '\0')

return (firstOcc);

if (src[i] == '\0')

return (-1);

i = firstOcc + 1;//while loop increment

j = 0;

}

}

//Program details -To find maximum number out of 3 using ternary operator

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<stdio.h>

int main(){

int a ,b , c ;

printf ("number1 ");

scanf("%d",&a);

printf ("number2 ");

scanf("%d",&b);

printf ("number3 ");

scanf("%d",&c);

printf("%d",((a>b)?((a>c)?a:c) : (b>c)?b:c ));

return 0 ;

}

//Program details - Find Greatest common divisor 0f two positive integer

#include<stdio.h>

int main()

{

int i, n1, n2, min, gcd, flag=0;

// get numbers

do{

printf("\n Enter 1St. number : ");

scanf("%d", &n1);

}

while(n1 < 0);

do{

printf("\n Enter 2nd. number : ");

scanf("%d", &n2);

}

while(n2 < 0);

// Get min. of two numbers

if (n1 <= n2) min = n1; else min = n2;

// loop for ith. term

for (i=2; i<=min; i++)

{

// Find GCD of two numbers

if ((n1 % i == 0) && (n2 % i == 0))

{

gcd = i;

flag = 1;

}

}

if (flag == 1){

printf ("\n GCD of %d and %d is %d ", n1, n2, gcd);

}else{

printf("\n GCD of %d and %d is not possible", n1, n2);

}

}

//Program details -To Calculate the Average of set of N numbers.

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<stdio.h>

int main()

{

int i, num; float numIn, avg, sum=0.0;

printf ("\n Enter how many numbers to be average");

scanf("%d", &num);

// Get numbers for num times.

for (i=0; i<num; i++){

printf ("\n Enter %d number : ", i+1);

scanf("%f", &numIn);

sum += numIn; // Calculate sum

}

// Calculate Average

avg = sum / num;

printf ("\n Average of %d number is = %.2f", num, avg);

}

//Program details -Reverse the given number

#include<stdio.h>

int main()

{

long int numIn, rem, revNum=0;

printf ("\n Enter 7 digit +ve number : ");

scanf("%ld", &numIn);

while (numIn > 0)

{

rem = numIn % 10;// find remainder

revNum = (revNum \* 10) + rem; // Reverse the num

numIn /= 10; // eliminate the unit digit

}

printf ("\n Reverse of number is : %ld ", revNum);

}

//Program details - Find sum of nth. series (1/1! + 2/2! + 3/3! +..... n/n!)

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<stdio.h>

int main()

{

int i, j, nT; long int fact; double sum = 0.0, tmp = 0.0; // initialize values

printf("\n Enter nth. term : ");

scanf("%d", &nT);

// loop for ith. term

for (i=1; i<=nT; i++){

fact = 1; // Must init fact before calculation

// calculate factorial of ith. term

for (j=1; j<i; j++){

fact \*= i;

}

// Calc ith. term

tmp = i / (double) fact;

// calc sum of nth. term

sum += tmp;

}

printf ("\n Sum of series is : %lf ", sum);

}

//Program details -To Calculate simple interest

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<stdio.h>

int main()

{

float pa, ri, ny, si;

printf ("\n Enter Principal Amt.");

scanf("%f", &pa);

printf ("\n Enter rate of Interest");

scanf("%f", &ri);

printf ("\n Enter No. of Years");

scanf("%f", &ny);

// Calculate simple interest

si = pa \* ri \* ny;

printf ("\n Simple interest = %.2f", si);

}

//Program details - Swap two numbers without using third variable.

#include<stdio.h>

int main()

{

float n1, n2;

printf ("\n Enter value for first number : ");

scanf("%f", &n1);

printf ("\n Enter value for second number : ");

scanf("%f", &n2);

// Swap above two numbers using math trick

n2 = n1 + n2;

n1 = n2- n1;

n2 = n2 - n1;

printf ("\n Swap values are ");

printf ("\n Now first number is : %f ", n1);

printf ("\n Now second number is : %f ", n2);

}

//Program details -To take input and store it in structure and then print info of student with maximum marks.

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<stdio.h>

struct student{

int rollno;

char name[40];

int marks;

} stu[3] ;

int find\_max\_index( struct student\* s ){

int j; int max\_index = 0; int max\_marks = 0;

for(j = 0 ; j < 3 ; j ++ ){

if (s[j].marks > max\_marks)

{

max\_marks = s[j].marks ;

max\_index = j ;

}

}

return max\_index ;

}

int main()

{

int i = 0, j = 0;

int max\_index = 0;

int max\_marks = 0;

struct student\* structptr = stu ;

for(i = 0 ; i < 3 ; i++ ){

printf("Enter roll number,name ,marks of student %d " ,i+1) ;

scanf("%d %s %d" ,&stu[i].rollno,&stu[i].name,&stu[i].marks) ;

}

max\_index = find\_max\_index(structptr);

printf("maximum scorer rollno - %d \n" ,stu[max\_index].rollno) ;

printf("maximum scorer name %s \n" ,stu[max\_index].name) ;

printf("maximum scorer marks %d " ,stu[max\_index].marks) ;

return 0 ;

}