

**KARPAGAM ACADEMY OF HIGHER EDUCATION**

(Deemed to be University Established Under Section 3 of UGC Act, 1956)

	SEMESTER VI	L T P C
15PHU611	PHYSICS PRACTICALS – VI	- - 5 3

ANY TEN EXPERIMENTS

1. Matrix multiplication using C-program
2. Runge-Kutta method using C-program
3. Simpson's rule – C program
4. Matrix addition using C program
5. Matrix Subtraction using C program
6. Largest among N numbers using C program
7. Fibonoi Series using C program
8. Ascending order using C program
9. Descending order using C program
10. Maclaurin series using C program
11. Poisson distribution using C program
12. Gaussian distribution using C program
13. Unix operating system

SYLLABUS:

LIST OF PROGRAMS:

- MATRIX MULTIPLICATION USING C-PROGRAM.
- RUNGE KUTTA METHOD USING C PROGRAM.
- NUMERICAL INTEGRATION USING SIMPSON'S 1/3 RULE.
- FIBONACCI SERIES USING C PROGRAM.
- MATRIX ADDITION USING C PROGRAM.
- MATRIX SUBTRACTION USING C PROGRAM.

PRG.NO:1

DATE:

MATRIX MULTIPLICATION

AIM:

To write a c program for multiplying a 3x3 matrix.

PROGRAM:

```
1. #include<stdio.h>
2. #include<stdlib.h>
3. int main(){
4. int a[10][10],b[10][10],mul[10][10],r,c,i,j,k;
5. system("cls");
6. printf("enter the number of row=");
7. scanf("%d",&r);
8. printf("enter the number of column=");
9. scanf("%d",&c);
10. printf("enter the first matrix element=\n");
11. for(i=0;i<r;i++)
12. {
13. for(j=0;j<c;j++)
14. {
15. scanf("%d",&a[i][j]);
16. }
17. }
18. printf("enter the second matrix element=\n");
19. for(i=0;i<r;i++)
20. {
21. for(j=0;j<c;j++)
```

```
22. {
23. scanf("%d",&b[i][j]);
24. }
25. }

26. printf("multiply of the matrix=\n");
27. for(i=0;i<r;i++)
28. {
29. for(j=0;j<c;j++)
30. {
31. mul[i][j]=0;
32. for(k=0;k<c;k++)
33. {
34. mul[i][j]+=a[i][k]*b[k][j];
35. }
36. }
37. }
38. //for printing result
39. for(i=0;i<r;i++)
40. {
41. for(j=0;j<c;j++)
42. {
43. printf("%d\t",mul[i][j]);
44. }
45. printf("\n");
46. }
47. return 0;
48. }
```

OUTPUT:

enter the number of row=3

enter the number of column=3

enter the first matrix element=

1 1 1

2 2 2

3 3 3

enter the second matrix element=

1 1 1

2 2 2

3 3 3

multiply of the matrix=

6 6 6

12 12 12

18 18 18

RESULT:

The above program is done successfully and the output is verified.

PRG.NO:2

DATE:

RUNGE KUTTA METHOD

AIM:

Write a C program to obtain the solution of numerical differential equation using Runge Kutta Method.

PROGRAM:

```
1. #include<stdio.h>
2. #include<math.h>
3. float f(float x,float y);
4. int main()
5. {
6. float x0,y0,m1,m2,m3,m4,m,y,x,h,xn;
7. printf("Enter x0,y0,xn,h:");
8. scanf("%f %f %f %f",&x0,&y0,&xn,&h);
9. x=x0;
10. y=y0;
11. printf("\n\nX\t\tY\n");
12. while(x<xn)
13. {
14. m1=f(x0,y0);
15. m2=f((x0+h/2.0),(y0+m1*h/2.0));
16. m3=f((x0+h/2.0),(y0+m2*h/2.0));
17. m4=f((x0+h),(y0+m3*h));
18. m=((m1+2*m2+2*m3+m4)/6);
19. y=y+m*h;
20. x=x+h;
21. printf("%f\t%f\n",x,y);
22. }
23. }
24. float f(float x,float y)
25. {
26. float m;
27. m=(x-y)/(x+y);
```

```
28. return m;  
29. }
```

OUTPUT:

```
Enter x0,y0,xn,h:0 2 2 0.5  
  
X          Y  
0.500000   1.621356  
1.000000   1.242713  
1.500000   0.864069  
2.000000   0.485426  
  
Process returned 16384 (0x4000)   execution time : 5.825 s  
Press any key to continue.  
  
codewithc.com
```

RESULT:

The above program is done successfully and the output is verified.

PRG.NO:3**DATE:****NUMERICAL INTEGRATION USING SIMPSON'S 1/3 RULE****AIM:**

Write a C program to evaluate the integral by using Trapezoidal rule.

PROGRAM:

```
1. #include<stdio.h>
2. #include<conio.h>
3. float f(float x)
4. {
5.     return(1/(1+x));
6. }
7. void main()
8. {
9.     int i,n;
10.    float x0,xn,h,y[20],so,se,ans,x[20];
11.    printf("\n Enter values of x0,xn,h: ");
12.    scanf("%f%f%f",&x0,&xn,&h);
13.    n=(xn-x0)/h;
14.    if(n%2==1)
15.    {
16.        n=n+1;
17.    }
18.    h=(xn-x0)/n;
19.    printf("\n Refined value of n and h are:%d %f\n",n,h);
20.    printf("\n Y values: \n");
21.    for(i=0; i<=n; i++)
22.    {
23.        x[i]=x0+i*h;
24.        y[i]=f(x[i]);
25.        printf("\n %f\n",y[i]);
26.    }
27.    so=0;
28.    se=0;
29.    for(i=1; i<n; i++)
30.    {
31.        if(i%2==1)
32.        {
33.            so=so+y[i];
```

```
34. }
35. else
36. {
37. se=se+y[i];
38. }

39. }
40. ans=h/3*(y[0]+y[n]+4*so+2*se);
41. printf("\n Final integration is %f",ans);

42. getch();
43. }
```

OUTPUT:

```
Enter values of x0,xn,h: 2 4 0.5
Refined value of n and h are:4 0.500000
Y values:
0.333333
0.285714
0.250000
0.222222
0.200000
Final integration is 0.510847
```

codewithc.com

RESULT:

The above program is done successfully and the output is verified.

PRG.NO:4

DATE:

FIBONACCI SERIES

AIM:

To Write a program of Fibonacci series using the c program.

PROGRAM:

```
1. #include<stdio.h>
2. int main()
3. {
4.     int n1=0,n2=1,n3,i,number;
5.     printf("Enter the number of elements:");
6.     scanf("%d",&number);
7.     printf("\n%d %d",n1,n2);//printing 0 and 1
8.     for(i=2;i<number;++i)//loop starts from 2 because 0 and 1 are already printed
9.     {
10.        n3=n1+n2;
11.        printf(" %d",n3);
12.        n1=n2;
13.        n2=n3;
14.    }
15.    return 0;
16. }
```

OUTPUT:

Enter the number of elements:15

0 1 1 2 3 5 8 13 21 34 55 89 144 233 377

RESULT:

The above program is done successfully and the output is verified.

PRG.NO:5

DATE:

MATRIX ADDITION

AIM:

To write a c program for adding two matrices.

PROGRAM:

```
1. #include<stdio.h>
2. int main() {
3. int i, j, mat1[10][10], mat2[10][10], mat3[10][10];
4. int row1, col1, row2, col2;
5. printf("\nEnter the number of Rows of Mat1 : ");
6. scanf("%d", &row1);
7. printf("\nEnter the number of Cols of Mat1 : ");
8. scanf("%d", &col1);
9. printf("\nEnter the number of Rows of Mat2 : ");
10. scanf("%d", &row2);
11. printf("\nEnter the number of Columns of Mat2 : ");
12. scanf("%d", &col2);
13. /* Before accepting the Elements Check if no of
14. rows and columns of both matrices is equal */
15. if (row1 != row2 || col1 != col2) {
16. printf("\nOrder of two matrices is not same ");
17. exit(0);
18. }
19. //Accept the Elements in Matrix 1
20. for (i = 0; i < row1; i++) {
21. for (j = 0; j < col1; j++) {
22. a. printf("Enter the Element a[%d][%d] : ", i, j);
23. b. scanf("%d", &mat1[i][j]);
24. }
```

```
23. }

24. //Accept the Elements in Matrix 2
25. for (i = 0; i < row2; i++)
26. for (j = 0; j < col2; j++) {
    a. printf("Enter the Element b[%d][%d] : ", i, j);
    b. scanf("%d", &mat2[i][j]);
27. }
28. //Addition of two matrices
29. for (i = 0; i < row1; i++)
30. for (j = 0; j < col1; j++) {
    a. mat3[i][j] = mat1[i][j] + mat2[i][j];
31. }
32. //Print out the Resultant Matrix
33. printf("\nThe Addition of two Matrices is : \n");
34. for (i = 0; i < row1; i++) {
35. for (j = 0; j < col1; j++) {
    a. printf("%d\t", mat3[i][j]);
36. }
37. printf("\n");
38. }
39. return (0);
40. }
```

OUTPUT:

Enter the number of Rows of Mat1 : 3

Enter the number of Columns of Mat1 : 3

Enter the number of Rows of Mat2 : 3

Enter the number of Columns of Mat2 : 3

Enter the Element a[0][0] : 1

Enter the Element a[0][1] : 2

Enter the Element a[0][2] : 3

Enter the Element a[1][0] : 2

Enter the Element a[1][1] : 1

Enter the Element a[1][2] : 1

Enter the Element a[2][0] : 1

Enter the Element a[2][1] : 2

Enter the Element a[2][2] : 1

Enter the Element b[0][0] : 1

Enter the Element b[0][1] : 2

Enter the Element b[0][2] : 3

Enter the Element b[1][0] : 2

Enter the Element b[1][1] : 1

Enter the Element b[1][2] : 1

Enter the Element b[2][0] : 1

Enter the Element b[2][1] : 2

Enter the Element b[2][2] : 1

The Addition of two Matrices is :

2 4 6

4 2 2

2 4 2

RESULT:

The above program is done successfully and the output is verified.

PRG.NO:6

DATE:

MATRIX SUBTRACTION

AIM:

To write a c program using matrix subtraction.

PROGRAM:

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

2. int main() {
3. int i, j, mat1[10][10], mat2[10][10], mat3[10][10];
4. int row1, col1, row2, col2;

5. printf("\nEnter the number of Rows of Mat1 : ");
6. scanf("%d", &row1);
7. printf("\nEnter the number of Cols of Mat1 : ");
8. scanf("%d", &col1);

9. printf("\nEnter the number of Rows of Mat2 : ");
10. scanf("%d", &row2);
11. printf("\nEnter the number of Columns of Mat2 : ");
12. scanf("%d", &col2);

13. /* Before accepting the Elements Check if no of
14. rows and columns of both matrices is equal */
15. if (row1 != row2 || col1 != col2) {
16. printf("\nOrder of two matrices is not same ");
17. exit(0);
18. }

19. //Accept the Elements in Matrix 1
20. for (i = 0; i < row1; i++) {
21. for (j = 0; j < col1; j++) {
```

```
        a. printf("Enter the Element a[%d][%d] : ", i, j);
        b. scanf("%d", &mat1[i][j]);
22. }
23. }

24. //Accept the Elements in Matrix 2
25. for (i = 0; i < row2; i++)
26. for (j = 0; j < col2; j++) {
        a. printf("Enter the Element b[%d][%d] : ", i, j);
        b. scanf("%d", &mat2[i][j]);
27. }

28. //Subtraction of two matrices
29. for (i = 0; i < row1; i++)
30. for (j = 0; j < col1; j++) {
        a. mat3[i][j] = mat1[i][j] - mat2[i][j];
31. }

32. //Print out the Resultant Matrix
33. printf("\nThe Subtraction of two Matrices is : \n");
34. for (i = 0; i < row1; i++) {
35. for (j = 0; j < col1; j++) {
        a. printf("%d\t", mat3[i][j]);
36. }
37. printf("\n");
38. }

39. return (0);
40. }
```

OUTPUT:

Enter the number of Rows of Mat1 : 3
Enter the number of Columns of Mat1 : 3

Enter the number of Rows of Mat2 : 3
Enter the number of Columns of Mat2 : 3

Enter the Element a[0][0] : 2
Enter the Element a[0][1] : 4
Enter the Element a[0][2] : 6
Enter the Element a[1][0] : 4
Enter the Element a[1][1] : 2
Enter the Element a[1][2] : 2
Enter the Element a[2][0] : 2
Enter the Element a[2][1] : 4
Enter the Element a[2][2] : 2

Enter the Element b[0][0] : 1
Enter the Element b[0][1] : 2
Enter the Element b[0][2] : 3
Enter the Element b[1][0] : 2
Enter the Element b[1][1] : 1
Enter the Element b[1][2] : 1
Enter the Element b[2][0] : 1
Enter the Element b[2][1] : 2
Enter the Element b[2][2] : 1

The Subtraction of two Matrices is:

1 2 3
2 1 1
1 2 1

RESULT:

The above program is done successfully and the output is verified.