

NAME:	Maureen Miranda
UID:	2022300060
SUBJECT	Operating system
EXPERIMENT NO :	2
DATE OF PERFORMANCE	1/02/24
DATE OF SUBMISSION	2/02/24
AIM:	<p>Create a scientific calculator program in C using static and dynamic linking. Create a separate file for each operation:</p> <ol style="list-style-type: none"> 1. Logarithmic operations (natural logarithm and log base 10) 2. Exponential operations (e^x and x^y) 3. Factorial operation ($n!$)
THEORY:	<p>Static Linking: The linker copies all library routines used in the program into executable images. As a result, it requires more memory space. As it does not require the presence of library on the system when it is running, so it is faster and more portable and there is less chance of error.</p> <p>Dynamic linking: Dynamic linking is performed during the run time. This linking is accomplished by placing the name of a shareable library in the executable image. It requires less memory space as multiple programs can share a single copy of the library.</p>

- Static linking

a. Create a C file for performing any complex operation (for example: finding the area of a cylinder, factorial of a number, prime factors of a number, etc)

```
C expo.c > power(int, int)
1 //Exponential operations (e^x and x^y)
2 #include<stdio.h>
3 #include<math.h>
4
5 int expo(int x)
6 {
7     return exp(x);
8 }
9
10 int power(int x,int y)
11 {
12     return pow(x,y);
13 }
```

```
C fact.c > fact(int)
1 //Factorial operation (n!)
2 #include<stdio.h>
3
4
5 int fact(int n)
6 {
7     int product =1;
8     for(int i=1;i<=n;i++)
9     {
10         product*=i;
11     }
12     return product;
13 }
14
```

```
> Users > dcmau > Downloads > C log.c
1 //Logarithmic operations (natural logarithm and log base 10)
2
3 #include<stdio.h>
4 #include<math.h>
5
6 int naturallog(int num)
7 {
8     int logarithm = log(num);
9     return logarithm;
10 }
11 int logten(int num)
12 {
13     int logarithm = log(num)/log(10);
14     return logarithm;
15 }
16
```

b. Create a header file (lib_mylib.h) for the library.

```
mylib.h > fact(int)
int fact(int n);
int expo(int x);
int power(int x,int y);
int naturallog(int num);
int logten(int num);
```

c. Create a driver program that uses the created library.

```
er.c > main()
#include<stdio.h>
#include<stdlib.h>
#include<stdbool.h>
#include "_mylib.h"
#include<math.h>

void main()
{
int a;

bool flag=true;
int num=0;
int num1=0;
while(flag)
{
    printf("\nselect from the below options\n1-natural log\n2-logtobase10\n3-exponential\n");
    scanf("%d",&a);
    switch(a)
    {
        case 1:
        {
            num=0;
            printf("natural log: enter number\n");
            scanf("%d",&num);
            printf("natural log= %d",naturallog(num));
            break;
```

```

}
case 2:
{
num=0;
printf("log to base 10:enter number\n");
scanf("%d",&num);
printf("natural log= %d",logten(num));
break;
}
case 3:
{
num=0;
printf("exponential :enter number\n");
scanf("%d",&num);
printf("exponential = %d",exp(num));
break;
}
case 4:
{
num=0;
num1=0;
printf("x^y:enter 2 numbers\n");
printf("one");
scanf("%d",&num);
printf("two");
scanf("%d",&num1);

```

```

53     printf(" %d , %d , pow(%d,%d) , /",
54     break;
55     }
56     case 5:
57     {
58     num=0;
59     printf("factorial; :enter number\n");
60     scanf("%d",&num);
61     printf("factorial = %d",fact(num));
62     break;
63     }
64     case 6:
65     {
66         flag=false;
67         break;
68     }
69 }
70 }
71
72
73 }

```

d. Creating the static library. Compiling the driver program and including the static library in it.

```

^C
(base) students@students-HP-280-G3-SFF-Business-PC:~/Desktop/static$ gcc -c expo.c-o expo.o
gcc: error: expo.c-o: No such file or directory
(base) students@students-HP-280-G3-SFF-Business-PC:~/Desktop/static$ gcc -c expo.c -o expo.o
(base) students@students-HP-280-G3-SFF-Business-PC:~/Desktop/static$ gcc -c log.c -o log.o
(base) students@students-HP-280-G3-SFF-Business-PC:~/Desktop/static$ gcc -c expo.c -o expo.o
(base) students@students-HP-280-G3-SFF-Business-PC:~/Desktop/static$ gcc -c fact.c -o fact.o
(base) students@students-HP-280-G3-SFF-Business-PC:~/Desktop/static$ ar rcs lib_mylib.a expo.o fact.o log.o
(base) students@students-HP-280-G3-SFF-Business-PC:~/Desktop/static$ gcc -c driver.c -o driver.o -lm
driver.c: In function 'main':
driver.c:41:28: warning: format '%d' expects argument of type 'int', but argument 2 has type 'double' [-Wformat=]
   41 |         printf("exponential = %d",exp(num));
      |                                ^~
      |                                |
      |                                double
      |                                int
      |                                %f
driver.c:53:20: warning: format '%d' expects argument of type 'int', but argument 2 has type 'double' [-Wformat=]
   53 |         printf("x^y = %d",pow(num,num1));
      |                        ^~
      |                        |
      |                        double
      |                        int
      |                        %f
(base) students@students-HP-280-G3-SFF-Business-PC:~/Desktop/static$ gcc -o driver driver.o -L. -l_mylib -lm

```

e. Executing the driver program.

```

      |                                int
      |                                %f
(base) students@students-HP-280-G3-SFF-Business-PC:~/Desktop/static$ gcc -o driver driver.o -L. -l_mylib -lm
(base) students@students-HP-280-G3-SFF-Business-PC:~/Desktop/static$ ./driver

```

```
3
exponential :enter number
52
exponential = 9603
select from the below options
1-natural log
2-logtobase10
3-exponential
4-power(x,y)
5-factorial
6-exit
4
x^y:enter 2 numbers
one3
two4
x^y = 812
select from the below options
1-natural log
2-logtobase10
3-exponential
4-power(x,y)
5-factorial
6-exit
5
factorial; :enter number
5
factorial = 120
select from the below options
1-natural log
2-logtobase10
3-exponential
4-power(x,y)
5-factorial
6-exit
6
```



```
3
exponential :enter number
52
exponential = 9603
select from the below options
1-natural log
2-logtobase10
3-exponential
4-power(x,y)
5-factorial
6-exit
4
x^y:enter 2 numbers
one3
two4
x^y = 812
select from the below options
1-natural log
2-logtobase10
3-exponential
4-power(x,y)
5-factorial
6-exit
5
factorial; :enter number
5
factorial = 120
select from the below options
1-natural log
2-logtobase10
3-exponential
4-power(x,y)
5-factorial
6-exit
6
```

2. Dynamic Linking: a. After creating the same C, header, and driver files as above, compile the C files. Creating a dynamic library and compiling the driver program using the dynamic library.

```

0
(base) students@students-HP-280-G3-SFF-Business-PC:~/Desktop/static$ gcc expo.c -c -fPIC -o expo.o -lm
(base) students@students-HP-280-G3-SFF-Business-PC:~/Desktop/static$ gcc fact.c -c -fPIC -o fact.o -lm
(base) students@students-HP-280-G3-SFF-Business-PC:~/Desktop/static$ gcc log.c -c -fPIC -o log.o -lm
(base) students@students-HP-280-G3-SFF-Business-PC:~/Desktop/static$ gcc -shared -o lib_mylib.so fact.o expo.o log.o
(base) students@students-HP-280-G3-SFF-Business-PC:~/Desktop/static$ gcc -shared -o lib_mylib.so fact.o expo.o log.o -lm
(base) students@students-HP-280-G3-SFF-Business-PC:~/Desktop/static$ export LD_LIBRARY_PATH=$PWD:$LD_LIBRARY_PATH
(base) students@students-HP-280-G3-SFF-Business-PC:~/Desktop/static$ gcc -L. -o driver driver.c -l_mylib -lm
driver.c: In function 'main':
driver.c:41:28: warning: format '%d' expects argument of type 'int', but argument 2 has type 'double' [-Wformat=]
  41 |     printf("exponential = %d",exp(num));
      |                        ~^      ~~~~~
      |                        |      |
      |                        |      double
      |                        int
      |                        %f
driver.c:53:20: warning: format '%d' expects argument of type 'int', but argument 2 has type 'double' [-Wformat=]
  53 |     printf("x^y = %d",pow(num,num1));
      |                ~^      ~~~~~
      |                |      |
      |                |      double
      |                int
      |                %f

```

b. Executing the driver program.

```

(base) students@students-HP-280-G3-SFF-Business-PC:~/Desktop/static$ ./driver
select from the below options
1-natural log
2-logtobase10
3-exponential
4-power(x,y)
5-exponential

```

select from the below options

- 1-natural log
- 2-logtobase10
- 3-exponential
- 4-power(x,y)
- 5-factorial
- 6-exit

1

natural log: enter number

23

natural log= 3

select from the below options

- 1-natural log
- 2-logtobase10
- 3-exponential
- 4-power(x,y)
- 5-factorial
- 6-exit

2

log to base 10:enter number

34

natural log= 1

select from the below options

- 1-natural log
- 2-logtobase10
- 3-exponential
- 4-power(x,y)
- 5-factorial
- 6-exit

3

exponential :enter number

44

exponential = 8125

select from the below options

- 1-natural log
- 2-logtobase10
- 3-exponential
- 4-power(x,y)

Conclusion: Successfully created a static/dynamic link library for finding the factorial, exponential and logarithm of a number and then tested this library through linuxid linker.