**THREADS**

**Read the syntax of following system call and proceed with the program**

**pthread\_create(),pthread\_join(),pthread\_exit(),pthread\_self(),pthread\_equal(),**

**/\*program to demonstrate use of threads\*/**

#include<stdio.h>

#include<string.h>

#include <pthread.h>

void \*myfunc (void \*mydata);

pthread\_t theThread;

main()

{

pthread\_t thread1;

char \*my1="first thread";

int ret1;

ret1=pthread\_create(&thread1,NULL,&myfunc,(void\*)my1);

printf("main function after pthread");

pthread\_join(thread1,NULL);

printf("first thread ret1=%d\n",ret1);

if(pthread\_equal(thread1,theThread))

{

printf("success");

}

}

void \*myfunc (void \*mydata)

{

char \*msg;

msg=(char\*)mydata;

int i;

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

{

printf("%s%d",msg,i);

sleep(1);

theThread=pthread\_self();

}

pthread\_exit(NULL);

}

**/\*program to demonstrate use of threads\*/**

#include<stdio.h>

#include<string.h>

#include <pthread.h>

void \*myfunc (void \*mydata);

main()

{

pthread\_t thread1,thread2;

char \*my1="first thread";

char \*my2="second thread";

int ret1,ret2;

ret1=pthread\_create(&thread1,NULL,&myfunc,(void\*)my1);

ret2=pthread\_create(&thread2,NULL,&myfunc,(void\*)my2);

printf("main function after pthread");

pthread\_join(thread1,NULL);

pthread\_join(thread2,NULL);

printf("first thread ret1=%d\n",ret1);

printf("second thread ret1=%d\n",ret2);

}

void \*myfunc (void \*mydata)

{

char \*msg;

msg=(char\*)mydata;

int i;

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

{

printf("%s%d",msg,i);

sleep(1);

}

return NULL;

}