

สมาชิก

Keerati Setthasuk 6733021521

Peerawit Seemamahawong 6733183521

Supanut Udompataikul 6733278521

ข้อ 1

```
source code
~ 4s cat 01.c
#include<stdio.h>
#include<pthread.h>
#include<unistd.h>
#include <string.h>
void* say_hello(void* data)
{
    char *str;
    str = (char*)data;
    while(1)
    {
        printf("%s\n",str);
        sleep(1);
    }
}
void main(int argc, char *argv[])
{
    if (argc < 3){
        printf("usage : thread_sol1 arg1 arg2\n");
        return;
    }
    pthread_t t1,t2;
    char buffer1[256];
    char buffer2[256];
    sprintf(buffer1, "This is thread %s\n", argv[1]);
    sprintf(buffer2, "This is thread %s\n", argv[2]);
    pthread_create(&t1,NULL,say_hello, (void*)buffer1);
    pthread_create(&t2,NULL,say_hello, (void*)buffer2);
    pthread_join(t1,NULL);
    pthread_join(t2,NULL);
}
```

output

```
< 0s @ ./thread_sol1
usage : thread_sol1 arg1 arg2

< 0s @ ./thread_sol1 cedt chula
This is thread cedt

This is thread chula

This is thread cedt

This is thread chula

This is thread cedt

This is thread chula

This is thread cedt

This is thread chula

This is thread cedt

This is thread chula
```

This is thread cedt

This is thread chula

This is thread cedt

This is thread cedt

This is thread chula

This is thread cedt

This is thread chula

This is thread cedt

This is thread chula

This is thread cedt

This is thread chula

□

- 751808 -

- 751809 -

- 751810 -

752097 - ./thread_sol1 cedt chula

- 752097 -

- 752098 -

- 752099 -

ข้อ 2

source code

```
[keerati@keerati-swiftsfg1672 04]$ cat 02.c
#include<stdio.h>
#include <unistd.h>
#include<pthread.h>
#include <sys/wait.h>
void* say_hello(void* data)
{
    char *str;
    str = (char*)data;
    printf("%s\n",str);
}
int main(){
    pid_t pid;
    pthread_t t1,t2;
    pid = fork();
    if (pid < 0) {
        printf("Error Create Child");
        return 0;
    }
    else if (pid == 0) {
        pthread_create(&t1, NULL, say_hello, "First thread from child process\n");
        pthread_create(&t2, NULL, say_hello, "Second thread from child process\n");
        pthread_join(t1, NULL);
        pthread_join(t2, NULL);
    }
    else {
        pthread_create(&t1, NULL, say_hello, "First thread from parent process\n");
        pthread_create(&t2, NULL, say_hello, "Second thread from parent process\n");
        pthread_join(t1, NULL);
        pthread_join(t2, NULL);
        wait(NULL);
    }
    return 0;
}
```

output

```
[keerati@keerati-swiftsfg1672 04]$ ./thread_proc
First thread from parent process

Second thread from parent process

First thread from child process

Second thread from child process

[keerati@keerati-swiftsfg1672 04]$ !!
./thread_proc
First thread from parent process

Second thread from parent process

Second thread from child process

First thread from child process

[keerati@keerati-swiftsfg1672 04]$ !!
./thread_proc
Second thread from parent process

First thread from parent process

First thread from child process

Second thread from child process
```