

دستور کار آزمایشگاه سیستم های عامل



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

آزمایشگاه سیستم های عامل

آرین محسنی

آبان 1403

دستور کار آزمایشگاه سیستم های عامل

تعریف مسئله:

مسئله Prefix Sum:

فرض کنید یک آرایه a داریم که در هر خانه اش عددی نوشته شده است. حال می خواهیم آرایه b بسازیم به طوری که:

$$b[i] = \text{sum}(a[0..i])$$

برای مثال اگر آرایه a برابر آرایه زیر باشد:

| $a[0]$ | $a[1]$ | $a[2]$ | $a[3]$ | $a[4]$ | $a[5]$ | $a[6]$ | $a[7]$ | $a[8]$ | $a[9]$ |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| ۱ | ۲ | ۱ | ۵ | ۰ | ۷ | ۳ | ۴ | ۲ | ۷ |

آرایه b ای که از a ساخته می شود برابر است با:

| $b[0]$ | $b[1]$ | $b[2]$ | $b[3]$ | $b[4]$ | $b[5]$ | $b[6]$ | $b[7]$ | $b[8]$ | $b[9]$ |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| ۱ | ۳ | ۴ | ۹ | ۹ | ۱۶ | ۱۹ | ۲۳ | ۲۵ | ۳۲ |

در این دستور کار می خواهیم با استفاده از چند thread محاسبات را به طور موازی

پیش ببریم

1) کد بخش بدون thread (پیچیدگی زمانی $O(n)$)

```
1  #include <stdio.h>
2  #define SIZE 10
3
4
5  int main(){
6
7      int array[SIZE];
8      int secondArray[SIZE];
9
10     printf("Enter %d number :",SIZE);
11     for (int i = 0; i < SIZE; i++)
12     {
13         scanf("%d",&array[i]);
14     }
15
16     for (int i = 0; i < SIZE; i++)
17     {
18         secondArray[i] = 0;
19     }
20
21     secondArray[0] = array[0];
22
23     for (int i = 1; i < SIZE; i++)
24     {
25         secondArray[i] = secondArray[i-1] + array[i];
26     }
27
28     for (int i = 0; i < SIZE; i++)
29     {
30         printf("%d\n",secondArray[i]);
31     }
32
33
34
35
36     return 0;
37 }
38
39
```

دستور کار آزمایشگاه سیستم های عامل

برنامه به صورت موازی و دو نخه به این شکل است

```
9  int array[SIZE];
10 int secondArray[SIZE];
11
12 void first(){
13     for (int i = 1; i < SIZE/2; i++)
14     {
15         secondArray[i] = secondArray[i-1] + array[i];
16     }
17 }
18 void second(){
19     for (int i = SIZE/2; i < SIZE; i++)
20     {
21         secondArray[i] = secondArray[i-1] + array[i];
22     }
23 }
24 }
25 int main(){
26
27     printf("Enter Number:\n");
28     for (int i = 0; i < SIZE; i++)
29     {
30         scanf("%d",&array[i]);
31     }
32
33     pthread_t thread_one;
34     pthread_t thread_two;
35
36     pthread_create(&thread_one, 0, first, 0);
37     pthread_create(&thread_two, 0, second, 0);
38
39     pthread_join(thread_one, 0);
40     pthread_join(thread_two, 0);
41
42     printf("After Two Threads\n");
43     for (int i = 0; i < SIZE; i++)
44     {
45         int tmp=0;
46         if(i < SIZE/2){
47             tmp+= array[i];
48             printf("%d\n",secondArray[i] + array[0]);
49         }
50         else
51             printf("%d\n", secondArray[i] + tmp + array[0]);
52     }
53
54     return 0;
55 }
```

اگر ارایه را دو بخش کنیم، باید مجموع بخش اول ارایه را به تمام المان های بخش دوم ارایه اضافه کنیم تا مشکلی به وجود نیاید.

دستور کار آزمایشگاه سیستم های عامل

برای 5000

```
Time taken for two-threaded execution: 0.000388 seconds
```

برای 50000

```
Time taken for two-threaded execution: 0.007501 seconds
```

برای 500000

```
Time taken: 3.132232 seconds
```