운영체제 – conditional variable

* **코드**

//

// Name : 2생산자 1소비자 동기화

// Date : 2019. 11. 22.

//

#include <stdio.h>

#include <pthread.h>

#include <ctype.h>

pthread\_cond\_t empty, fill;

pthread\_mutex\_t mutex;

#define MAX 10

#define loops 51

#define MAXLOOPS 102

// 공유버퍼 및 공유변수

**char** buffer[MAX];

**int** count = 0, num1 = 0, num2 = 0;

**char** c = 'A';

// A ~ Z 그리고 a ~ z 까지 버퍼에 넣는 함수

**void** put(){

// Z 부터 a까지 건너뛰기

**if**(c=='Z')

c+=7;

buffer[num2] = c;

num2++;

c++;

count++;

**if**(num2==10)

num2=0;

}

// 1 ~ 50 까지 버퍼에 넣는 함수

**void** put2(**int** value){

buffer[num2] = value;

num2++;

count++;

**if**(num2==10)

num2=0;

}

// 화면에 출력하는 함수

**void** get(){

**if**(isalpha(buffer[num1])) // 문자인지 검사

printf("%c\t", buffer[num1]);

**else** // 아니면 숫자

printf("%d\t", buffer[num1]);

num1++;

count --;

**if**(num1==10){

printf("\n");

num1 = 0;

}

}

// 쓰레드 1 : A ~ Z / a ~ z

**void** \*producer(**void** \*arg){

**int** i;

**for**(i = 0; i<loops; i++){

pthread\_mutex\_lock(&mutex);

**while**(count==MAX)

pthread\_cond\_wait(&empty, &mutex);

put();

pthread\_cond\_signal(&fill);

pthread\_mutex\_unlock(&mutex);

}

**return** **NULL**;

}

// 쓰레드 2 : 1 ~ 50

**void** \*producer2(**void** \*arg){

**int** i;

**for**(i = 0; i<loops; i++){

pthread\_mutex\_lock(&mutex);

**while**(count==MAX)

pthread\_cond\_wait(&empty, &mutex);

put2(i);

pthread\_cond\_signal(&fill);

pthread\_mutex\_unlock(&mutex);

}

**return** **NULL**;

}

// 쓰레드 3 : 화면 출력

**void** \*consumer(**void** \*arg){

**int** i;

**for**(i = 0; i<MAXLOOPS; i++){

pthread\_mutex\_lock(&mutex);

**while**(count == 0 && i!=101)

pthread\_cond\_wait(&fill, &mutex);

get();

pthread\_cond\_signal(&empty);

pthread\_mutex\_unlock(&mutex);

}

**return** **NULL**;

}

**int** main(**int** argc, **const** **char** \* argv[]) {

pthread\_t p1, p2, c1;

// mutex, cond 초기화

pthread\_mutex\_init(&mutex, **NULL**);

pthread\_cond\_init(&empty, **NULL**);

pthread\_cond\_init(&fill, **NULL**);

// 쓰레드 생성

pthread\_create(&p1, **NULL**, producer, **NULL**);

pthread\_create(&p2, **NULL**, producer2, **NULL**);

pthread\_create(&c1, **NULL**, consumer, **NULL**);

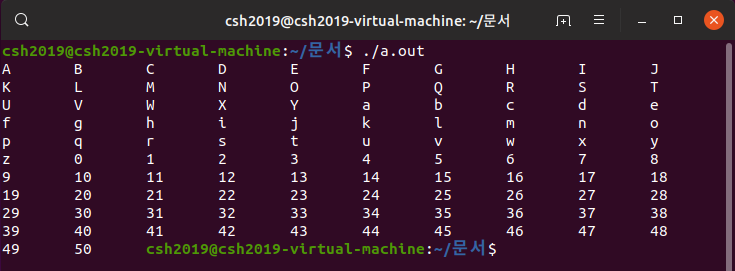
// 쓰레드 대기

pthread\_join(c1, **NULL**);

**return** 0;

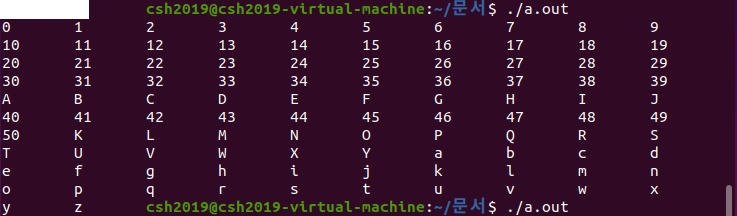
}

* **컴퓨터, 실내, 모니터이(가) 표시된 사진

  자동 생성된 설명첫번째 실행결과**
* **두번째 실행결과**
* **실내이(가) 표시된 사진

  자동 생성된 설명세번째 실행결과**
* **컴퓨터, 실내, 모니터이(가) 표시된 사진

  자동 생성된 설명네번째 실행결과**
* **다섯번째 실행결과**

****