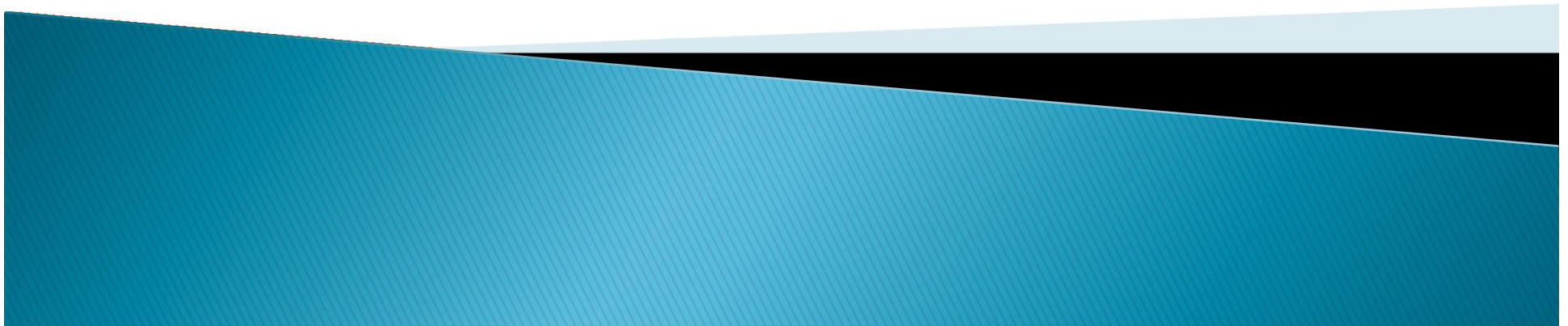


프로그래밍랩 - 7주 실행시간

2018-1학기
정인환교수



Lab 7-1:현재시간

```
C:\Windows\system32\cmd.exe
now = 1426154853
ctime() : Thu Mar 12 19:07:33 2015
2015년 3월 12일 목요일 19시 7분 33초
```

▶ time.h 의 함수들 이용 방법

- `time_t now;`
- `time(&now);` // 1970년 이후 1초 단위로 증가
- `char *ctime(&now);` // 시간을 문자열로 반환
 - `printf("ctime() : %s", ctime(&now));`
- `struct tm *localtime(const time_t *);`
- `struct tm *tp = localtime(&now);`

```
struct tm { /* a broken-down time */
    int tm_sec; /* seconds after the minute: [0 - 60] */
    int tm_min; /* minutes after the hour: [0 - 59] */
    int tm_hour; /* hours after midnight: [0 - 23] */
    int tm_mday; /* day of the month: [1 - 31] */
    int tm_mon; /* months since January: [0 - 11] */
    int tm_year; /* years since 1900 */
    int tm_wday; /* days since Sunday: [0 - 6] */
    int tm_yday; /* days since January 1: [0 - 365] */
    int tm_isdst; /* daylight saving time flag: <0, 0, >0 */
};
```

Lab 7-2:움직이는시계

C:\Windows\system32\cmd.exe

2015년 3월 12일 목요일 20시 5분 55초

C:\Windows\system32\cmd.exe

2015년 3월 12일 금요일 20시 6분 45초

- ▶ 무한 Loop 돌면서 ctime()을 출력한다면

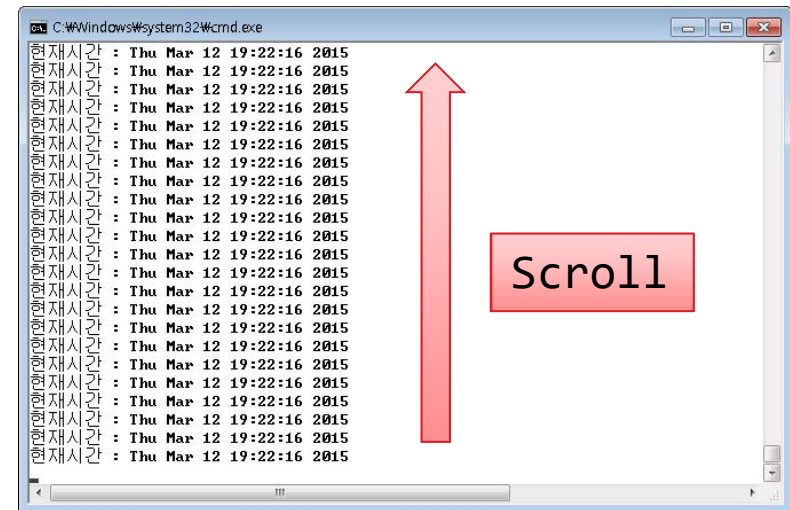
```
while (1) {
    time(&now);
    printf("현재시간 : %s", ctime(&now));
}
```

- 화면 Scroll 문제
- 같은 시간 표시

- ## ▶ 해결 방법

- struct tm 0|00 (Lab1)
- Hint : \r 0|00

```
while (1)
    printf("%d\r", time(NULL));
```



C:\Windows\system32\cmd.exe

1426156308



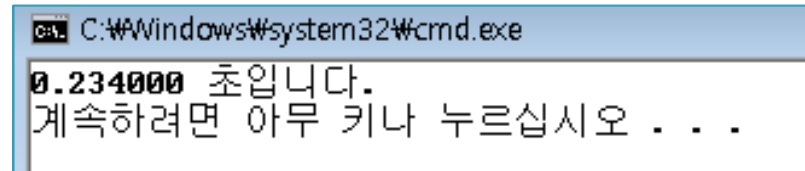
C:\Windows\system32\cmd.exe

1426156327

Lab 7-3 실행 시간 측정

- ▶ clock_t clock() 함수 이용 : millisec 단위로 측정

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
void main(void )
{
    clock_t start, finish;
    double  duration;
    int i, j, k;
    start = clock();
    // 수행시간을 측정하고자 하는 코드....
    for (i=1;i<=10000;i++)
        for (j=1;j<10000;j++)
            k++;
    finish = clock();
    duration = (double)(finish - start) / CLOCKS_PER_SEC;
    printf("%f 초입니다.\n", duration);
}
```



Lab 7-4: 시작/종료시간 표시 및 시간측정

- ▶ Lab7-1,7-3 이용
 - ctime(), localtime(), clock(), struct tm
- ▶ 약 20초 실행 시간이 필요한 프로그램

```
for (i=1;i<=100000;i++)
    for (j=1;j<100000;j++)
        k++;
```
- ▶ 시작/종료 시간, 실행시간 모두 표시

- ctime(), clock() 이용 →

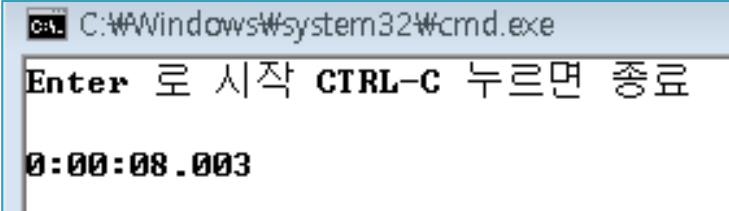
- localtime(&tm) 이용 →

```
C:\Windows\system32\cmd.exe
시작 : Thu Mar 12 21:20:55 2015
종료 : Thu Mar 12 21:21:18 2015
실행시간 = 22.403000 초입니다.

시작 : 21:21:18
종료 : 21:21:40
실행시간 = 22.418000 초입니다.
계속하려면 아무 키나 누르십시오 . . .
```

Lab 7-5: 스톱워치

- ▶ Enter 키를 누르면 시계가 시작
 - 경과시간 계산
 - `start = clock()`
 - `now = clock()`
 - 경과시간 `duration = now - start`
 - Milisec 단위이므로
 - 시, 분, 초, milisec 로 환산하고
 - 화면에 표시한다
 - `hh:mm:ss.nnn`
 - Enter 로 시작, CTRL-C 누르면 종료
 - 또는 아무키나 누르면 종료
 - `conio.h` 의 `kbhit()` 이용
 - `kbhit()==1` 이면 key가 눌린것임1



```
C:\Windows\system32\cmd.exe
Enter 로 시작 CTRL-C 누르면 종료
0:00:08.003
```

Lab7-ACM: 피보나치 응용

- ▶ 모든 양의 정수는 Fibonacci 수의 조합으로 표현 가능하다
 - 0 1 1 2 3 5 8 13 21 34 55 89
 - 100 =
 - 3 + 8 + 89 ($f(4) + f(6) + f(11)$)
 - 1 + 2 + 8 + 89
 - 3 + 8 + 34 + 55
- ▶ 문제
 - 양의 정수($n \leq 1,000,000,000$)가 주어지면 합이 같아지는 최소 개수의 피보나치 수 열 구하기
- ▶ Algorithm: find_fibos(int n)// 재귀함수로 작성 가능 $n=100, 11, 3$
 - $n==1$ 이면 1을 출력하고 return
 - $f \leq n$ 인 피보나치 최대값 f 를 찾고 // $n=100$ $f=89$, $n=11$ $f=8$, $n=3$ $f=3$
 - $f = \text{find_max_fibo}(n)$
 - $f==n$ 이면 f 를 출력 하고 return
 - $(n-f)>0$ 이면 $n=(n-f)$ 에 대해 다시 피보나치 조합을 재귀함수로 찾는다

Input	Output
5	3 8 89
100	1 55 144
200	1 34 377 987 10946
12345	3 13 987
1003	1 5 13 89 233 1597 28657 75025 196418 514229 5702887 24157817 267914296 701408733
1000000000	

Lab7-ACM-1: 피보나치 응용

- ▶ `int find_max_fibo(int n)`
 - Fibonacci 수열 중 $f \leq n$ 인 최대값 f 를 찾는 함수
 - 재귀함수로 작성하면 시간이 많이 소요됨
 - 반복적인 방법을 사용하여야 함 (for 이용)

The 37th Annual ACM
International Collegiate Programming Contest
Asia Regional – Daejeon
Nationwide Internet Competition



Problem D

피보나치(Fibonacci)

피보나치 수 f_k 는 $f_k = f_{k-1} + f_{k-2}$ 로 정의되며 초기값은 $f_0 = 0$ 과 $f_1 = 1$ 이다. 양의 정수는 하나 혹은 그 이상의 서로 다른 피보나치 수들의 합으로 나타낼 수 있다는 사실은 잘 알려져 있다.

하나의 양의 정수에 대한 피보나치 수들의 합은 여러 가지 형태가 있다. 예를 들어 정수 100은 $f_4 + f_6 + f_{11} = 3 + 8 + 89$ 또는 $f_1 + f_3 + f_6 + f_{11} = 1 + 2 + 8 + 89$, 또는 $f_4 + f_6 + f_9 + f_{10} = 3 + 8 + 34 + 55$ 등으로 나타낼 수 있다. 이 문제는 하나의 양의 정수를 최소 개수의 서로 다른 피보나치 수들의 합으로 나타내는 것이다.

하나의 양의 정수가 주어질 때, 피보나치 수들의 합이 주어진 정수와 같게 되는 최소 개수의 서로 다른 피보나치 수들을 구하라.

입력(Input)

입력 데이터는 표준입력을 사용한다. 입력은 T 개의 테스트 데이터로 구성된다. 입력의 첫 번째 줄에는 테스트 데이터의 수를 나타내는 정수 T 가 주어진다. 각 테스트 데이터에는 하나의 정수 n 이 주어진다. 단, $1 \leq n \leq 1,000,000,000$.

출력(Output)

출력은 표준출력을 사용한다. 하나의 테스트 데이터에 대한 해를 하나의 줄에 출력한다. 각 테스트 데이터에 대해, 피보나치 수들의 합이 주어진 정수에 대해 같게 되는 최소수의 피보나치 수들을 증가하는 순서로 출력한다.

다음은 네 개의 테스트 데이터에 대한 입력과 출력의 예이다.

입력 예제(Sample Input)	출력 예제(Output for the Sample Input)
4	3 8 89
100	1 55 144
200	1 34 377 987 10946
12345	3 13 987
1003	

ICPC 2012 Asia Regional – Daejeon Nationwide Internet Competition Problem D: 피보나치