REPORT

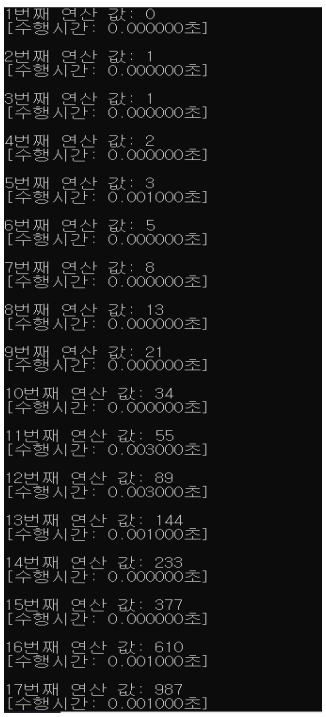
과 제 명: HW#5 피보나치 수열의 계산

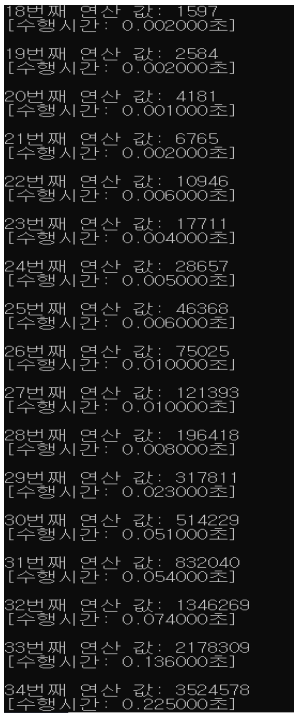
과목명	자료구조 01분반
교수명	송오영
학번	20206319
학과	소프트웨어학부 소프트웨어전공
이름	김가연

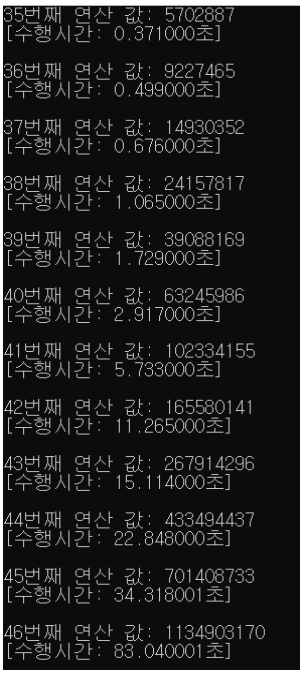
[Recursive call를 이용한 코드]

```c
#include <stdio.h>
#include <time.h>

long fib(int n) {
	if (n < 2) {
		return n;
	}
	return(fib(n - 1) + fib(n - 2));
}

int main() {
	clock_t ct;
	for (int i = 0; i < 46; i++) {
		ct = clock();
		long f = fib(i);
		printf("%d번째 연산 값: %d\n", i + 1, f);
		ct = clock() - ct;
		printf("[수행시간: %f초]\n", (float)ct / CLOCKS_PER_SEC);
		printf("\n");
	}
}
```

[출력 결과]



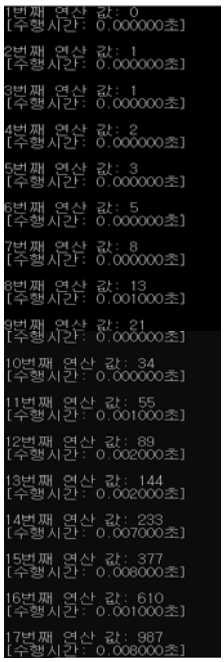


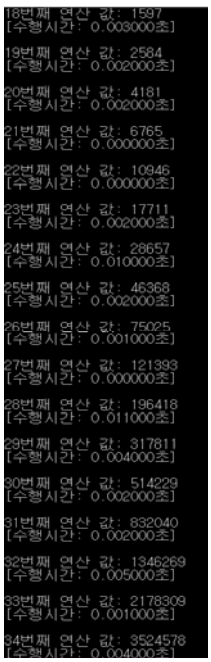


[Recursive call with memoization을 이용한 코드]

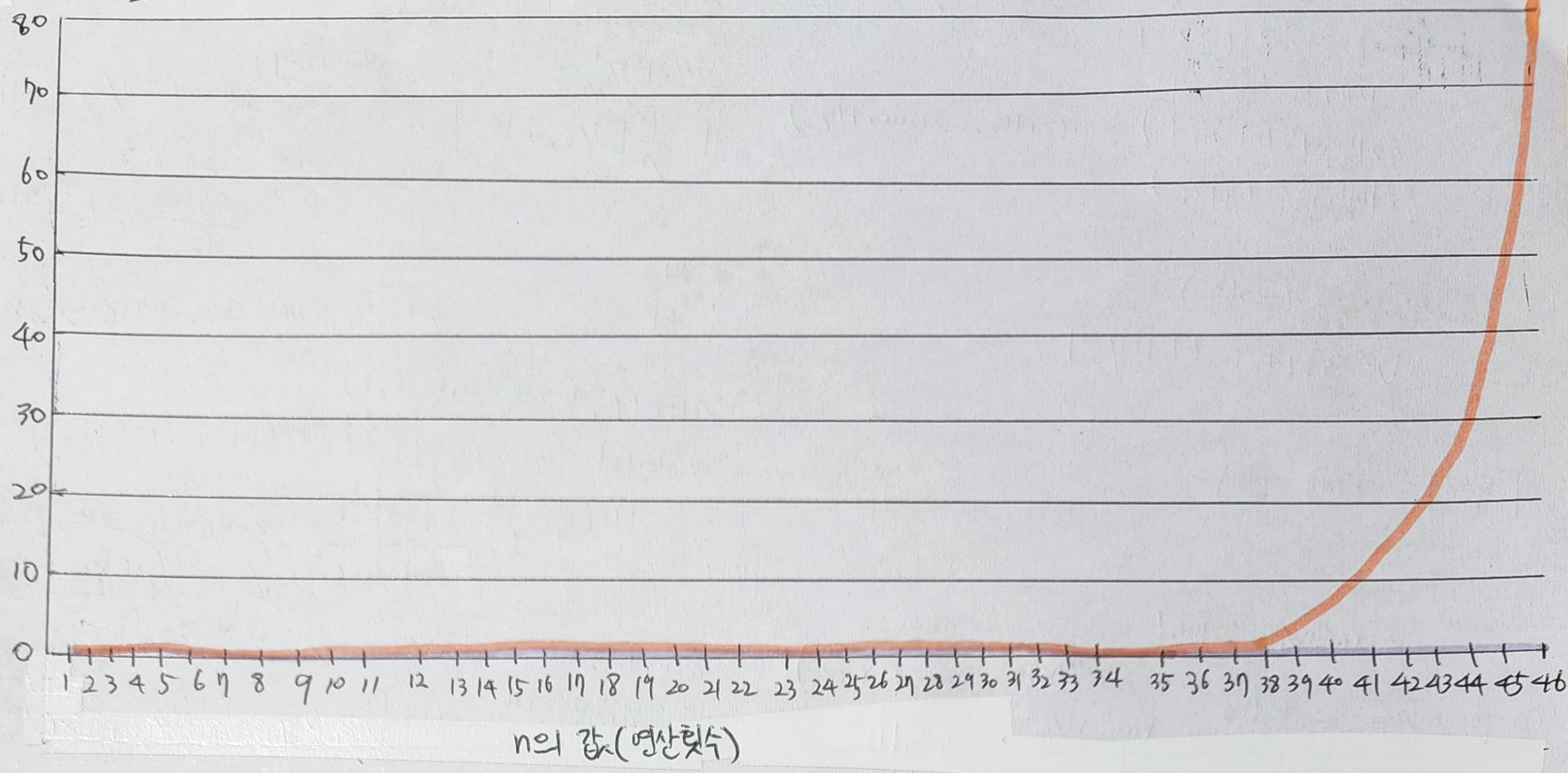
```c
#include <stdio.h>
#include <time.h>

long Save[999999];
long fib(int n) {
	if (n < 2) {
		return n;
	}
	if (Save[n] != 0) return Save[n];
	Save[n] = fib(n - 1) + fib(n - 2);
	return Save[n];
}

int main() {
	clock_t ct;
	for (int i = 0; i < 46; i++) {
		ct = clock();
		long f = fib(i);
		printf("%d번째 연산 값: %d\n", i + 1, f);
		ct = clock() - ct;
		printf("[수행시간: %f초]\n", (float)ct / CLOCKS_PER_SEC);
		printf("\n");
	}
}
```

[출력 결과]





[그래프 및 결론]



Recursive call with memoization은 0초대에 프로그램을 수행하였다. Recursive call은 Recursive call with memoization과 함께 프로그램 수행시간이 비슷하게 이루어졌으나, 38번째부터는 1초대 이상의 시간이 걸리고, 42번째에는 대략 10초대의 시간이 걸리기 시작하며 그 이후로는 기하급수적으로 프로그램 실행시간이 증가한다.

memoization은 컴퓨터 프로그램이 동일한 계산을 수행할때, 이전에 계산한 값을 메모리에 저장함으로써 동일한 계산의 반복 수행을 제거하여 실행 속도를 빠르게 하는 기술이다.

피보나치 수열의 값을 구하기 위한 fib함수가 재귀적으로 실행되면서 같은 값에 대하여 계속해서 반복하게 된다.

이에 따라, Recursive call은 연산 횟수가 증가함에 따라 같은 값을 또다시 계산하게 되면서 시간이 오래 걸리게 되는 것이고, memoization을 사용하면 불필요한 반복의 과정을 줄였기에 빠르게 수행될 수 있었다.

통상적으로 시간이 짧게 걸리는 프로그램은 작성해진 코드로 이루어진 좋은 프로그램이라 한다. 불필요한 과정없이 원하는 일이 수행되기 때문이다. 이와 같이, 반복적인 일을 수행하고 처리할 때에는 memoization 사용이 시간을 단축시키기에 좋으며 이로 보대로 좋은 프로그램은 만드는데 많은 사용이 이루어 질 것 같다.

처음에는 수행 횟수를 5회로 지정하였다. 그런데, 43번째의 피보나치 수열값이 너무 커져 음수가 나오는 에러가 발생하였고, 실행시간 또한 앞선 횟수들보다 매우 길어지게 됨에 따라 결과를 얻는데 시간이 꽤 소요되었다. 이에 따라 수행횟수는 42회로 선정하였다. 피보나치 수열의 값은 무한으로 커질수 있기에 일단 최대하게 큰 숫자를 표현하는 long 타입으로 결과를 설정하였으며, 수행시간은 float와 double 타입 모두 사용해본후 굳이 double: 타입이 아니어도 괜찮기에 float 타입으로 설정하였다.