

컴퓨터 알고리즘과 실습

3월 2주차



학번 : 2016110056

학과 : 불교학부

이름 : 박승원

날짜 : 2017년 3월 8일

문제 1 (30점)

- 당신에게 1과 N 사이의 정수 중 하나를 제외한 N-1 개의 서로 다른 정수가 주어져 있다. 선형시간 $O(N)$ 안에 제외된 정수를 찾는 알고리즘을 구현해 보시오.
 1. 선형시간에 답을 구할 아이디어를 써보시오. (15점)
 2. 유사코드를 써보시오. (5점)
 3. 프로그래밍 언어로 구현하고 코드를 제출하시오. (5점)
 4. 2번에서 작성한 코드로 (5,6,2,4,9,10,8,7,1) 을 입력하였을 때 출력을 화면 캡처하여 제출 하시오. (5점)

힌트: 숫자의 합을 이용

1. 모든 합을 구한뒤, 1부터 N까지의 합에서 뺀다.

2. $i = 1$ to N : $sum += ar[i]$
return $N(N+1)/2 - sum$

3.

```
// 2016110056 박승원
#include<iostream>
using namespace std;

int main() {
    int ar[] = {5,6,2,4,9,10,8,7,1};
    int sum = 0;
    for(int i=0; i<9; i++) sum += ar[i];
    cout << 55 - sum << endl;
}
```

```
4 전체 int co[5]{0,};
while(M > 0) for(int i=0; i<5; i++) while(M >= c[i]) {
----- 문제 2번 실행을 시작합니다. (5,6,2,4,9,10,8,7,1)을 입력하였을
./2.x ----- 때 출력을 화면 캡처하여 제출 하시오. (5점)
1 0 1 1 0 ----- 문제 2번 실행을 종료합니다.
g++ 1.cpp -o 1.x -g -fmax-errors=1 -lm -std=c++14
head 1.cpp
//2016110056 박승원: 숫자의 합을 이용
#include<iostream>
using namespace std;

int main() {
    int ar[] = {5,6,2,4,9,10,8,7,1};
    int sum = 0;
    for(int i=0; i<9; i++) sum += ar[i];
    cout << 55 - sum << endl;
}
----- 문제 1번 실행을 시작합니다. -----
./1.x
3 ----- 문제 1번 실행을 종료합니다. -----
```

4.

문제 2 (70점)

- 강의시간에 GreedyBetterChange 는 $c=(25,20,10,5,1)$ 일때 incorrect 한 결과를 돌려주는 것을 살펴 보았다. 어떤 M 에 대해서도 GreedyBetterChange가 정당한(correct)한 결과를 돌려주기 위해서 c 에 새로운 크기의 동전 c_6 를 더해보자.
 - GreedyBetterChange 의 유사코드를 적어보아라. (5점)
 - GreedyBetterChange 를 프로그래밍 언어로 구현해 보고 $M=40$, $c=(25,20,10,5,1)$ 에 대한 결과를 제출하시오. (10점)
 - BruteForceChange 의 유사코드를 적어보아라. (5점)
 - BruteForceChange 를 프로그래밍 언어로 구현해 보고 $M=40$, $c=(25,20,10,5,1)$ 에 대한 결과를 제출하시오. (10점)
 - 어떤 M 에 대해서도 GreedyBetterChange가 correct한 결과를 주기위해 필요한 c_6 는 무엇인가? 가능한 값이 여러개일 경우 다 적으시오. (20점)
 - $c=(25,20,10,5,1)$ 를 이용하였을 경우와 5번에서 구한 c_6 를 포함한 6개의 동전을 이용하였을 경우 각각에 대해 M 을 1부터 100까지 대입해보면서 2번에서 구현한 GreedyBetterChange 를 프로그램의 결과와 4번에서 구현한 BruteForceChange 의 결과를 비교하고 서로 다를 경우 이를 출력하는 프로그램을 작성해서 코드와 출력물(화면 캡처)을 제출하시오. (20점)

```
1. while M < 0 :
    for i 0 to N :
        while M > c[i] : M -= c[i], coin++
```

2.

```
// 2016110056 박승원
#include<iostream>
using namespace std;

int main()
{
    int M = 40;
    int c[5] = {25,20,10,5,1};
    int co [5]{0,};
    while(M > 0) for(int i=0; i<5; i++) while(M >= c[i]) {
        M -= c[i];
        co[i]++;
    }
    for(int i=0; i<5; i++) cout << co[i] << ' ';
}
```

```
96 : 3 1 0 0 0 1
97 : 3 1 0 0 0 2
98 : 3 1 0 0 0 3
99 : 3 1 0 0 0 4
100 : 4 0 0 0 0 0
----- 문제 6번 실행을 종료합니다. -----
g++ 2.cpp -o 2.x -g -fmax-errors=1 -lm -std=c++14
head 2.cpp
//2016110056 박승원
#include<iostream>
using namespace std;

int main()
{
    int M = 40;
    int c[5] = {25,20,10,5,1};
    int co[5]{0,};
    while(M > 0) for(int i=0; i<5; i++) while(M >= c[i]) {
----- 문제 2번 실행을 시작합니다. -----
./2.x
1 0 1 1 0 ----- 문제 2번 실행을 종료합니다. -----
```

3. select a coin

M -= coin

if M != 0 : repeat selecting

4.

```
// 2016110056 박승원
#include<iostream>
using namespace std;

int M = 40;
int c[5] = {25,20,10,5,1};
int cc[5] = {0,}; // coin count
int re[5] = {0,};
```

```

int minc = 100;

int select (int m) {
    M -= c[m];
    cc[m]++;
    if (!M) { // 중단 조건
        int sum = 0;
        for(int i=0; i<5; i++) sum += cc[i]; //count coin
        if(sum < minc) {
            minc = sum;
            for(int i=0; i<5; i++) re[i] = cc[i]; //copy to result
        }
    }
    for(int i=0; i<5; i++) if(M >= c[i]) select (i); // 브루트 포스로 선택
    M += c[m];
    cc[m]--;
}

int main()
{
    for(int i=0; i<5; i++) select (i);
    for(int i=0; i<5; i++) cout << re[i] << ' ';
}

```

```

4 전체 cout << 55 - sum << endl;
}
----- 문제 1번 실행을 시작합니다. (5,6,2,4,9,10,8,7,1) 을 입력하였을
./1.x
3
----- 문제 1번 실행을 종료합니다. -----
g++ 3.cpp -o 3.x -g -fmax-errors=1 -lm -std=c++14
head 3.cpp
//2016110056 박승원
#include<iostream>
using namespace std;

int M = 40;
int c[5] = {25,20,10,5,1};
int cc[5] = {0,}; //coin count
int re[5] = {0,};
int minc = 100;

----- 문제 3번 실행을 시작합니다. -----
./3.x
0 2 0 0 0 ----- 문제 3번 실행을 종료합니다. -----

```

5. 15, 왜냐하면 한 코인의 가치의 두배를 만족시키는 자신보다 높은 값의 코인과 작은 값의 코인의 조합이 존재해야 한다.

6. // 2016110056 박승원

```

#include<iostream>
using namespace std;

int rr [2][100][6]; // to compare two result
const int c[6] = {25,20,15,10,5,1};
int co [6]{0,}; // coint count
int re[6] = {0,}; // result for recursive function
int M;
int minc = 100; //

void select (int m) {
    M -= c[m];
    co[m]++;
    int sum = 0;
    for(int i=0; i<6; i++) sum += co[i]; // count coin
    if(sum > minc) return;
    if (!M) { // 종단 조건
        int sum = 0;
        for(int i=0; i<6; i++) sum += co[i]; // count coin
        if(sum < minc) {
            minc = sum;
            for(int i=0; i<6; i++) re[i] = co[i]; // copy to result
        }
    }
    for(int i=0; i<6; i++) if(M >= c[i]) select (i); // 브루트 포스로 선택
    M += c[m];
    co[m]--;
}

int main()
{
    for(int k = 1; k<=100; k++) {
        M = k;
        while(M > 0) for(int i=0; i<6; i++) while(M >= c[i]) {
            M -= c[i];
            co[i]++;
        }
        for(int i=0; i<6; i++) rr [0][k-1][i] = co[i];
        for(int i=0; i<6; i++) co[i] = 0;
    }

    for(int k=1; k<=100; k++) {
        M = k;
        for(int i=0; i<6; i++) select (i);
    }
}

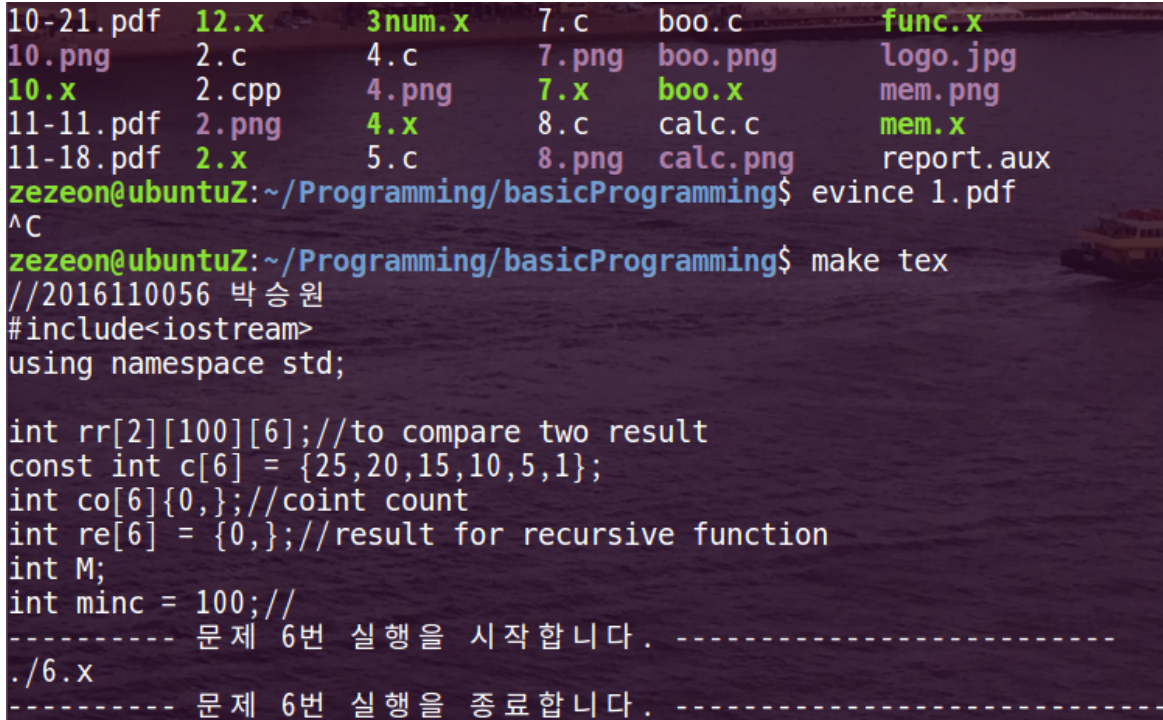
```

```

    for(int i=0; i<6; i++) {
        rr [1][k-1][i] = re[i];
        co[i] = 0;
    }
    minc = 100;
}

for(int i=0; i<100; i++) for(int j=0; j<6; j++)
    if(rr [0][i][j] != rr [1][i][j]) cout << i << endl;
}

```



```

10-21.pdf  12.x      3num.x      7.c      boo.c      func.x
10.png     2.c          4.c          7.png     boo.png     logo.jpg
10.x       2.cpp        4.png        7.x       boo.x       mem.png
11-11.pdf  2.png        4.x          8.c       calc.c      mem.x
11-18.pdf  2.x          5.c          8.png     calc.png    report.aux
zezeon@ubuntuZ:~/Programming/basicProgramming$ evince 1.pdf
^C
zezeon@ubuntuZ:~/Programming/basicProgramming$ make tex
//2016110056 박승원
#include<iostream>
using namespace std;

int rr[2][100][6]; //to compare two result
const int c[6] = {25,20,15,10,5,1};
int co[6]{0,}; //coint count
int re[6] = {0,}; //result for recursive function
int M;
int minc = 100; //
----- 문제 6번 실행을 시작합니다 . -----
./6.x
----- 문제 6번 실행을 종료합니다 . -----

```

모든 경우에 동일하므로 아무런 출력이 발생하지 않았다.