9월 1주차 스터디 2조 팀별 보고서

-정은이, 송예인, 김응철-

<1>

백준 1182 부분수열의 합

김응철

```java
import java.util.Scanner;

public class 부분수열의합_1182 {

    static int N,S,cnt=0;
    static int[] arr;
    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        N = sc.nextInt();
        S = sc.nextInt();

        arr = new int[N];

        for(int n=0;n<N;n++) {
            arr[n] = sc.nextInt();
        }

        subset(0,0);

        if(S==0) cnt--;

        System.out.println(cnt);
    }

    public static void subset(int target,int sum) {

        if(target==N) {

            if(sum==S)
                cnt++;

            return;
        }

        subset(target+1,sum);
        subset(target+1,sum+arr[target]);
    }
```

송예인

```java
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    N = sc.nextInt();
    S = sc.nextInt();
    arr = new int[N];
    for(int i=0;i<N;i++) {
        arr[i] = sc.nextInt();
    }
    // input end
    check = new boolean[N];

    subset(0);//현재 인덱스

    System.out.println(count);
}

//전형적인 부분집합 구하기 문제
//각각의 인덱스에 대해 선택될 기회와, 선택 안될기회 모두 준다.
//최종적으로 부분집합에 인덱스가 없는 경우를 제외하고 목적합인 S와 같다면
private static void subset(int idx) {
    if(idx == N) {
        int total = 0;
        boolean notallfalse = false;
        for(int n=0;n<N;n++) {
            if(check[n]) {
                total += arr[n];
                notallfalse = true;
            }
        }
        if(total == S && notallfalse) count++;
        return;
    }

    if(!check[idx]) {
        check[idx] = true;
        subset(idx+1);
        check[idx] = false;
        subset(idx+1);
    }
}
```

정은이

```java
public class Java_1182 {
    static int[] list;
    static int S, count;

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int N = sc.nextInt();
        S = sc.nextInt();
        list = new int[N];
        boolean[] isSelected = new boolean[N];


        for (int i = 0; i < N; i++) {
            list[i] = sc.nextInt();
        }
        subset(isSelected,0);
        System.out.print(count);

    }

    static void subset(boolean[] isSelected, int cnt) {
        if (cnt >= isSelected.length) {
            int sum = 0, n = 0;
            for (int i = 0; i < isSelected.length; i++) {
                if (isSelected[i]) {
                    sum += list[i];
                    n++;
                }
            }
            if (sum == S && n>0) count ++;
            return;
        }
        isSelected[cnt] = true;
        subset(isSelected, cnt + 1);
        isSelected[cnt] = false;
        subset(isSelected, cnt + 1);
    }
```

<2>

백준 10971외판원 순회

김응철

```java
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        N = sc.nextInt();
        map = new int[N+1][N+1];
        check = new boolean[N+1];
        dis = new int[N+1];
        for(int n=1;n<=N;n++) {
            for(int m=1;m<=N;m++) {
                map[n][m] = sc.nextInt();
            }
            dis[n] = Integer.MAX_VALUE;
        }
        // input end
        // 시작점 1로 설정
        // 시작점으로 다시 돌아와야 하기에 check true 는 안해줌

        result = Integer.MAX_VALUE;
        dis[1] = 0;
        check[1] = true;
        dfs(1,1,0);


        System.out.println(result);


    }

    //
    private static void dfs(int idx, int count, int price){
        if(count==N) {
            if(map[idx][1]==0) return;//마지막 점과 출발점이 연결되어 있지 않은경우
            result = result<price+map[idx][1]?result:price+map[idx][1];
            return;
        }

        //현재 idx에서 인접노드들 dfs 돌리기
        for(int i=1;i<=N;i++) {
            if(map[idx][i]!=0 && !check[i]) {
                check[i] = true;
                int origin = dis[i];
                dis[i] = dis[idx]+map[idx][i];
                dfs(i,count+1,dis[i]);
                check[i] = false;
                dis[i] = origin;
            }
        }

    }
```

송예인

```java
import java.util.*;
import java.io.*;

public class 외판원순회2_10971 {
    static int n, map[][], answer;
    static boolean visit[];

    public static void main(String[] args) throws Exception {
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        n = Integer.parseInt(br.readLine());
        StringTokenizer stz;
        map = new int[n+1][n+1];
        visit = new boolean[n+1];
        answer = Integer.MAX_VALUE;

        for(int i = 1; i <= n; i++) {
            stz = new StringTokenizer(br.readLine());
            for(int j = 1; j <= n; j++)
                map[i][j] = Integer.parseInt(stz.nextToken());
        }

        for(int i = 1; i <= n; i++) {
            visit[i] = true;
            dfs(i, i, 0);
            visit[i] = false;
        }

        System.out.println(answer);
    }

    public static void dfs(int start, int line, int sum) {
        boolean finish = true;
        for(int i = 1; i <= n; i++)
            if(!visit[i])
                finish = false;

        if(finish && map[line][start] != 0) {
            answer = Math.min(answer, sum+map[line][start]);
            return;
        }
```

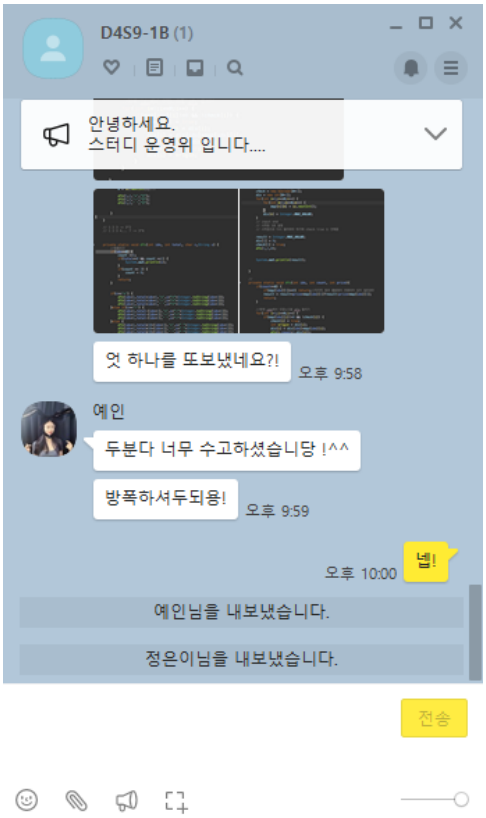
정은이

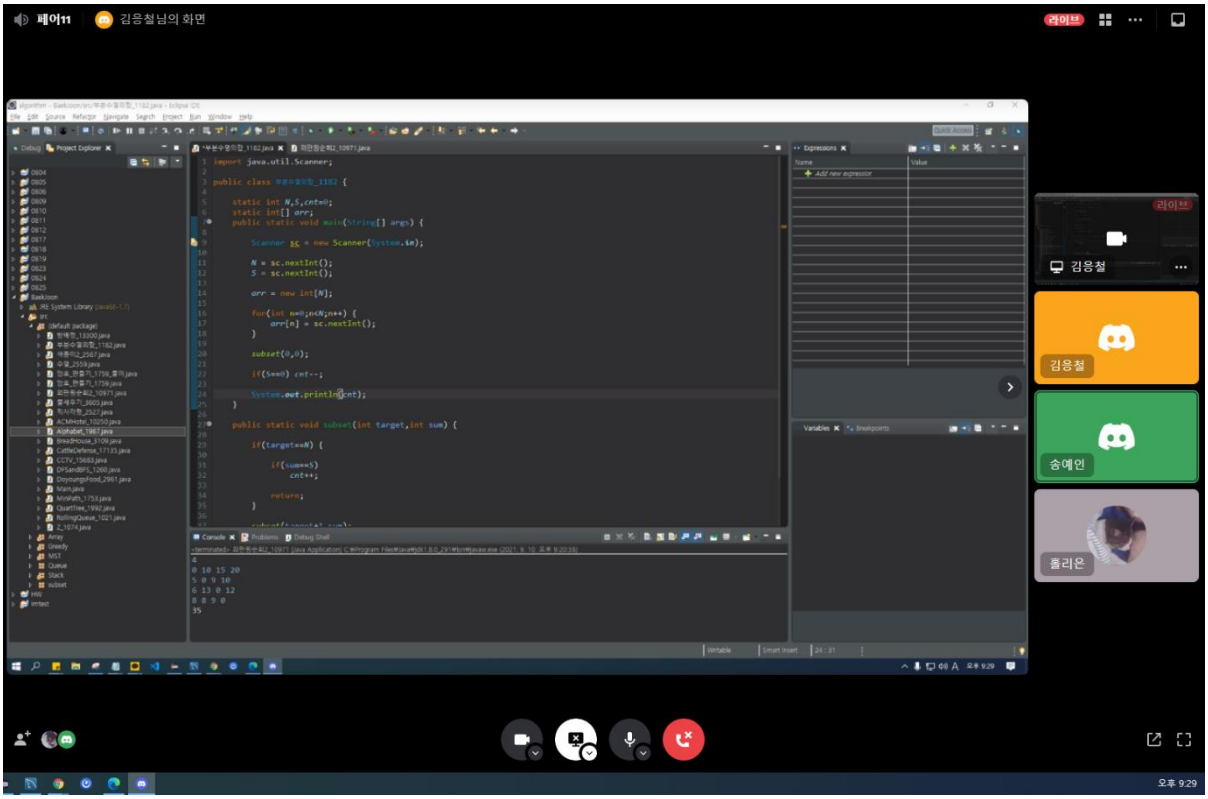
```java
public class Java_10971 {
    static int[][] adjMatrix;
    static int result = Integer.MAX_VALUE;

    public static void main(String[] args) throws IOException {
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        int N = Integer.parseInt(br.readLine());
        adjMatrix = new int[N][N];
        boolean[] add = new boolean[N];
        StringTokenizer st = null;

        for (int i = 0; i < N; i++) {
            st = new StringTokenizer(br.readLine());
            for (int j = 0; j < N; j++) {
                adjMatrix[i][j] = Integer.parseInt(st.nextToken());
            }
        }
        for (int i = 0; i < N; i++) {
            Arrays.fill(add, false);
            dfs(i, i, 0, 0, add);
        }
        System.out.print(result);
    }

    static void dfs(int start, int current, int cnt, int sum, boolean[] add) {
        if (cnt >= add.length) {
            result = Math.min(result, sum);
            return;
        }

        if (cnt == add.length - 1) {
            if ( adjMatrix[current][start] != 0 && !add[start]) {
                dfs(start, current, cnt + 1, sum + adjMatrix[current][start], add);
            }
        } else {
            for (int i = 0; i < add.length; i++) {
                if (adjMatrix[current][i] != 0 && !add[i] && i != start) {
                    add[i] = true;
                    dfs(start, i, cnt + 1, sum +  adjMatrix[current][i], add);
                    add[i] = false;
                }
            }
```

백준 74900 만들기

송예인

```java
    static int count;

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int TC = sc.nextInt();
        for(int tc=0;tc<TC;tc++) {
            N = sc.nextInt();//3

            dfs(1,1,'+',"1");
            dfs(1,1,'-',"1");
            dfs(1,1,' ',"1");


        }
    }

    // 1 2 3 -> 3^2
    // 1 2 3 4... 7 -> 3^6


    private static void dfs(int idx, int total, char c,String s) {
        //탈출조건
        if(idx==N) {
            count +=1;
            if(total==0 && count ==1) {
                System.out.println(s);
            }
            if(count == 3) {
                count = 0;
            }
            return;
        }


        if(c=='+') {
            dfs(idx+1,total+idx+1,'+',s+"+"+Integer.toString(idx+1));
            dfs(idx+1,total+idx+1,'-',s+"+"+Integer.toString(idx+1));
            dfs(idx+1,total+idx+1,' ',s+"+"+Integer.toString(idx+1));
        }else if(c=='-') {
            dfs(idx+1,total-(idx+1),'+',s+"-"+Integer.toString(idx+1));
            dfs(idx+1,total-(idx+1),'-',s+"-"+Integer.toString(idx+1));
            dfs(idx+1,total-(idx+1),' ',s+"-"+Integer.toString(idx+1));
        }else {
            dfs(idx+1,total*10+(idx+1),'+',s+" "+Integer.toString(idx+1));
            dfs(idx+1,total*10+(idx+1),'-',s+" "+Integer.toString(idx+1));
            dfs(idx+1,total*10+(idx+1),' ',s+" "+Integer.toString(idx+1));
        }

    }
}
```


D4S9-1B (1)
안녕하세요.
스터디 운영위 입니다....
엇 하나를 또보냈네요?!
오후 9:58
예인
두분다 너무 수고하셨습니당 !^^
방폭하셔두되용!
오후 9:59
오후 10:00
넵!
예인님을 내보냈습니다.
정은이님을 내보냈습니다.
전송


페어11
김응철님의 화면
라이브
김응철
송예인
홀리은