

정국경 모각코 4주차 문제풀이

주석에 문제풀이 달겠습니다.

<https://www.acmicpc.net/problem/14501>

```
/*
14501/퇴사/BOG
0ms
60m
*/

#include <iostream>
using namespace std;

int t[17];
int p[17];

int main()
{
    cin.tie(0); ios::sync_with_stdio(0);

    int n; cin >> n;

    // 입력은 헛갈리지 않기 위해 1~n 까지 받음
    for (int i=1; i<=n; ++i)
    {
        cin >> t[i] >> p[i];
    }

    int dp[18] = {0, };

    int result = 0;

    /*
```

dp[i] : i~n일에 벌 수 있는 금액의 최댓값 이라 하자.

뒤에서 앞으로 오며 dp array를 업데이트 한다.

만약 오늘의 일을 마무리할 수 있다면($i + t[i] \leq n + 1$)

현재까지의 금액 최댓값(dp[i + 1])

vs 오늘의 금액(p[i])과 일을 끝낸 뒤의 금액(dp[i + t[i]])의 합을 비교

마무리 불가 하다면

현재까지의 금액 최댓값(dp[i + 1])을 저장한다.

이때 오늘 일 마무리 가능 조건이 n 이하가 아닌 n+1 이하인 이유는,

만약 마지막 날 5일에 1일 걸리는 일이 있다면 $5 + t[5] = 6$ 이고 완수 가능하다.

따라서 n + 1 이하이면 완수 가능한 일인 것이다.

*/

```
for (int i=n; i>0; --i)
```

```
{
```

```
    // 완수 가능한 일
```

```
    if (i + t[i] <= n + 1)
```

```
    {
```

```
        dp[i] = max(dp[i + 1], p[i] + dp[i + t[i]]);
```

```
    }
```

```
    // 완수 불가능한 일
```

```
    else
```

```
    {
```

```
        dp[i] = dp[i + 1];
```

```
    }
```

```
}
```

```
// for (int i=0; i<=n; ++i)
```

```
// {
```

```
//     cout << "dp[" << i << "] = " << dp[i] << '\n';
```

```
//     result = max(result, dp[i]);
```

```
// }
```

```
cout << dp[1]; // 1 ~ n일 간에 벌 수 있는 금액의 최댓값
```

```
}
```