

2. 소스코드 함수별 설명

```
#include <stdio.h>
#include <iostream>
#include <string.h>
#include <stdlib.h>

void roundrobin(int job[], int fin[], int cycle[], int time_q ) //round robin 함수
{
    int num = 0; //변수 num은 각 작업의 완전한 종료시간을 알기 위함
    while(1)
    {
        for (int i = 0; i < 4; i++)
        {
            if (cycle[i] > time_q) //Time quantum보다 cpu cycle이 크면 Time quantum만큼 일단 실행함
            {
                cycle[i] = cycle[i] - 4; // 남은 작업을 구함
                num = num + 4; //현재까지의 작업 총 진행 시간
            }
            else if (cycle[i] > 0 && cycle[i] <= time_q) // cycle[i]의 값이 time quantum보다 작거나 같고 0보다 큰 경우
                // 0인 경우에는 다음 작업으로 건너 뛰므로 포함시키지 않았다.
            {
                num = num+ cycle[i]; // 작업만큼 더해준다
                fin[i] = num; //작업i 종료 시간
                cycle[i] = 0; //0인 경우에는 다음 작업으로 넘어가기 위함
            }
        }
    } //for문 종료
```

2. 소스코드 함수별 설명

```
if (cycle[0] == 0 && cycle[1] == 0 && cycle[2] == 0 && cycle[3] == 0) break; //이 조건이 만족 할 때까지 while문을 실행한다  
} //while문 종료
```

```
for (int i = 0; i < 4; i++) // 각 작업의 종료시간 출력  
{  
    printf("\nfin[%d]=%d \t", i, fin[i]);  
}  
}
```

```
int main() //main 함수
```

```
{  
    int time_q; //time quantum  
  
    printf("Time Quantum을 입력하시오 : ");  
    scanf_s("%d", &time_q);  
  
    int job[4] = { 0,1,2,3 }; // 각 도착시간은 0,1,2,3초 < 일찍 도착한 순서  
    int fin[4] = {}; //각 작업이 완료된 시간  
    int cycle[4] = { 8,4,9,5 }; // 각 작업의 CPU 사이클 = 8,4,9,5  
    int turnaround[4] = {}; // 작업 완료 시간- 작업 처음 도착 시간,, fin-job
```

2. 소스코드 함수별 설명

```
roundrobin(job, fin, cycle, time_q); //round robin 함수 호출
```

```
for (int i = 0; i < 4; i++) //turnaround 출력
```

```
{  
    turnaround[i] = fin[i] - job[i];  
    printf("\n\njob%d의 turnaround = %d\n", i, turnaround[i]);  
}
```

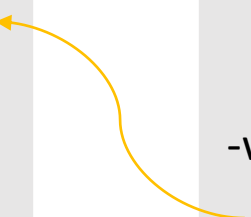
```
float sum_turnaround = 0; //average turnaround를 구하기 위한 변수
```

```
for (int i = 0; i < 4; i++)
```

```
{  
    sum_turnaround = sum_turnaround + turnaround[i];  
}  
printf("\nAverage Turnaround= %.3lf", sum_turnaround / 4);  
}
```

2. 소스코드 함수별 설명

Main 함수

- 1 - Time Quantum 입력
 - 도착 시간, CPU 사이클 지정
 - roundrobin 함수를 호출
 - 3 - Turnaround를 구하고 출력
 - 평균 turnaround값을 출력
- 

Roundrobin 함수

- 2 - num변수를 선언해
:작업 종료시간을 알기 위함
 - While문의 조건이
만족할 때 까지
For문 실행
 - while문이 종료되면 각 작업의
종료시간을 구하여 출력