201707297 채승민 HW\_11 과제

2023.05.28

public static int workloadRTAforEDF(double T, double C, double D, double L) {  
 int N = (((int) L / (int) T)); //𝑁\_𝑖 (𝑅\_𝑘^𝑛 )=⌊(𝑅\_𝑘^𝑛)/𝑇\_𝑖 ⌋  
 return N \* (int) C + Math.*min*((int) C, (int) L - (int) N \* (int) T); //𝐼\_𝑘^𝑖(𝑅\_𝑘^𝑛)=𝑁\_𝑖 (𝑅\_𝑘^𝑛 )∙𝐶\_𝑖+min⁡(𝐶\_𝑖,𝑅\_𝑘^𝑛−𝑁\_𝑖 (𝑅\_𝑘^𝑛)∙𝑇\_𝑖)  
}

int N = (((int) L / (int) T)); 을 통해 을 구한 후,

()= 에 적용시켜

N \* (int) C + Math.*min*((int) C, (int) L - (int) N \* (int) T); 을 반환하는

workloadRTAforEDF 함수 생성.

public int RTATestforEDF\_Multi()  
{  
 for (int k=0; k<pInfo.numTask; k++)  
 {  
 Task task\_k = pInfo.tasks.get(k);  
 int interval= (int)task\_k.execTime;  
 boolean pass=false;  
  
 while (interval<= (int)task\_k.Deadline && pass==false)  
 {  
 int sum=0;  
 for (int i=0; i<pInfo.numTask; i++)  
 {  
 Task task\_i = pInfo.tasks.get(i);  
 if (i!=k)  
 {  
 sum+=Math.*min*(*workloadRTAforEDF*(task\_i.Period, task\_i.execTime, task\_i.Deadline, interval),(int)task\_k.Deadline - (int)task\_k.execTime + 1);  
 }  
 }  
 int new\_interval = sum/pInfo.numProcessor+(int)task\_k.execTime;  
 if (interval>=new\_interval)  
 pass=true;  
 else  
 interval = new\_interval;  
 }  
 if (pass==false)  
 return 0;  
 }  
 return 1;  
}

if (i!=k){  
 sum+=Math.*min*(*workloadRTAforEDF*(task\_i.Period, task\_i.execTime, task\_i.Deadline, interval),(int)task\_k.Deadline - (int)task\_k.execTime + 1);  
}

🡪 i!=k 인 경우의 모든 을 sum에 더해 준다.

int new\_interval = sum/pInfo.numProcessor+(int)task\_k.execTime;

🡪 이후 프로세서의 개수로 나눠주고 task k 의 exectime을 더한 후,

if (interval>=new\_interval)  
 pass=true;  
 else  
 interval = new\_interval;

에서 과 비교를 하며 갱신해 나간다.

위 같은 과정을 통해 식을 만족하여

tk 가 schedulable 한지 알 수 있다.