운영체제 Lab01

2015140124

전자공학과

진우빈

**1. 프로그램에 사용되는 구조**

//Task의 정보를 담는 구조체: 태스크이름, 실행시간, 주기가 저장된다

**struct Task{**

**string name;**

**int computation\_time, period;**

**};**

//차트 데이터 구조체: 태스크의 이름, 해당 시간 방문 여부가 저장된다.

**struct Chartdata {**

**string name;**

**bool flag;**

**};**

**vector<Task> Ptasks**; //주기 태스크

**vector<Task> APtasks;** // 비주기 태스크 APtasks에서의 period는 도착시간이라고 가정

얼마나 많은 태스크들이 수행되는 지는 입력을 통해 받아야 알 수 있기 때문에 Vector를 사용 하여 태스크 정보를 입력받으면서 메모리를 할당하였습니다.

**Chartdata\* BackgroundChart;** //Background Aps에서 담기는 정보

**Chartdata\* PollingChart;** //Polling Server에서 담기는 정보

Background APS에서, Polling Server에서 각각 수행될 태스크는 다르기 때문에 따로 배열을 만들었습니다. 그리고 모든 주기 태스크들의 HyperPeriod를 계산하여 HyperPeriod만큼 태스크를 넣어줄 것이기 때문에 동적할당으로 생성하였습니다.

**int NumOfPTask, NumOfApTask;** // 각 태스트 들의 갯수

**int HyperPeriod = 1;** // 최소공배수 개산할 때 필요

**int Polling\_Capacity, Poliing\_Period;** //폴링서버의 Capacity와 주기

//최대 태스크 갯수 10개, 최대 Hyperbolic Period = 100

**int Gant[11][100];**

간트차트로 출력하기 위해서 만든 배열입니다. 행은 태스크, 열은 시간 입니다. BackgroundChart, PollingChart에 있는 데이터들을 Gant의 행에 맞추어 옮긴다음 Gant에서는 해당 시점에 데이터가 있으면 '='을 출력하여 대략적인 태스크 진행 상황을 볼 수 있습니다.

하지만 터미널에서 시행되므로 모니터가 크지 않는 한 HyperPeriod를 최대 80정도밖에 보여줄 수 없습니다.

**2. 사용 함수**

**1) void Input()**

사용자에게서 주기 태스크, 비주기 태스크의 갯수와 정보를 입력받는 함수입니다.

**2) void CalcLCM()**

주기 태스크의 HyperPeriod 즉, 태스크들의 최소공배수를 계산하는 함수입니다.

**3) void print\_table(vector<Task> v, int n)**

입력받은 태스크 정보를 순서별로 보여주는 함수입니다.

**4) void print\_gantt\_chart()**

Background, Polling server에서의 Task들이 어떤 시점에 들어가는지 계산이 되면 Gant 배열로 옮겨서 출력합니다.

**5) void CalcBackground()**

동적할당된 Background 배열에 주기. 비주기 태스크를 조건에 맞추어 집어넣습니다. 그리 고 그 때의 비주기 태스크의 평균지연시간을 계산합니다.

**6) void CalcPolling()**

동적할당된 Polling배열에 주기. 비주기 태스크를 조건에 맞추어 집어넣습니다. 그리 고 그 때의 비주기 태스크의 평균지연시간을 계산합니다.

**7) bool cmp(Task a, Task b)**

입력받은 주기, 비주기 태스크를 각각 주기가 짧은 순, 도착시간이 짧은 순으로 Sorted하기 위한 함수입니다.

**3. 실행 화면**

**프로그램의 주요 기능들 만을 캡쳐하였습니다.**

**<Periodic Task, Aperiodic Task의 정보 입력받기 : 갯수, Computation time, Period>**



**<입력받은 Task의 정보들을 Table 출력>**



테이블 출력 후 Polling Server에서의 Capacity와 주기를 입력받습니다.

**<Background, Polling 배열에 넣어진 태스크 정보를 토대로 간트차트 출력>**



**간트차트 출력과 동시에 계산된 평균 대기 시간을 출력합니다.**

**출력 이후에 0입력 시 프로그램이 종료되고 0제외 숫자 입력 시 계속 진행합니다.**