

**Struttura Dati:**

typedef char Tstringa[DIM];

typedef struct

{

Tstringa nomeProcesso;

int tempo;

} Tprocesso;

typedef struct Tnodo

{

Tprocesso processo;

struct Tnodo \*next;

} Tnodo;

typedef Tnodo\* TnodoPtr;

typedef struct

{

TnodoPtr testa;

TnodoPtr coda;

} Tlista;

typedef Tlista array[5];

* **SPIEGAZIONE DELLA SCELTA ADOTTATA PER LA STRUTTURA DATI:**

Questa struttura dati, costituita da un array di code differenziate per priorità, ci permette di ottimizzare l’efficienza dell’algoritmo poiché non è necessario effettuare un inserimento ordinato del processo scorrendo una lista singola.

* **GESTIONE DELLA CODA:**

La coda è un tipo di lista che viene gestita tramite due puntatori, uno alla testa ed uno alla coda.

Ogni volta che viene generato un nuovo processo, di priorità “p”, viene accodato alla lista corrispondente tramite il puntatore all’ultimo elemento.

* **GESTIONE DEL LAVORO:**

Il lavoro verrà svolto da tutto il gruppo riunito.