

Università degli Studi di Roma “La Sapienza”
Facoltà di Ingegneria – Corso di Laurea Specialistica in Ingegneria Informatica
Corso di Metodi Formali nell’Ingegneria del Software
Prof. Toni Mancini

Esercizio **E.II.20080703**

versione del 6 luglio 2008

Si consideri il seguente problema:

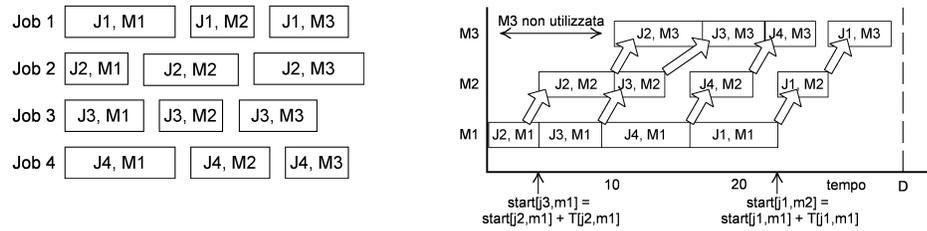
Un’industria che produce lastre di marmo deve produrre N lastre diverse. Il processo di produzione di ogni singola lastra attraversa in sequenza M fasi distinte (ad es., taglio, pulizia, infine lucidatura). In ogni fase è necessario uno tra M diversi macchinari (ad es., sega, compressore, lucidatrice). Tuttavia, a causa della possibile diversa grandezza delle diverse lastre da produrre, i tempi necessari nelle singole fasi sono diversi da lastra a lastra. Ogni macchinario può processare solo una lastra alla volta.

L’obiettivo è quello di assegnare alle diverse fasi di lavorazione delle diverse lastre un istante di inizio, in modo tale che tutte le lastre vengano lavorate completamente (tutte le fasi vengono eseguite) e correttamente (nell’ordine corretto) entro un certo tempo massimo, detto *deadline*.

Questo problema è una esemplificazione del seguente problema astratto di scheduling, una variante del più generale *Shop Scheduling*.

Siano dati:

- un insieme di N job $\{j_1, \dots, j_N\}$;
- un insieme di M macchine $\{m_1, \dots, m_M\}$, su cui i job devono essere processati sempre nella stessa sequenza (l’ i -esima fase di ogni job va eseguita sulla macchina m_i , dopo che sia terminata la fase $i - 1$, e deve terminare prima che inizi la fase $i + 1$);
- una deadline D (intero positivo);
- una matrice $(N \times M)$ T di tempi di processamento di ogni job su ogni macchina ($T[j, m]$ indica il numero di istanti di tempo (intero tra 1 e D) necessari alla macchina m per processare il job j).



(a) Una istanza con 4 job e 3 macchine. Il tempo di processamento di una fase di un job è dato dalla larghezza del relativo rettangolo.

(b) Una possibile soluzione.

Figura 1: Shop Scheduling.

Si vuole calcolare l'istante di inizio di tutte le fasi di processamento (su ogni macchina) di tutti i job (quindi un insieme di valori $start[j, m] \in 1..D$ per ogni $j \in 1..N$ e $m \in 1..M$) in modo che:

- La fase i -esima di ogni job venga eseguita sulla macchina i (per ogni $i \in [1..M]$);
- Ogni macchina operi su un solo job alla volta;
- La k -esima fase di lavorazione di ogni job ($k > 1$) non inizi prima che sia terminata la $(k - 1)$ -esima;
- Tutti i job siano completati al tempo D .

In Figura 1 viene mostrata una possibile istanza del problema insieme ad una possibile soluzione.

Si richiede di codificare un programma in un qualsivoglia linguaggio di programmazione ad alto livello che, presa in input un'istanza del problema (ovvero dei valori per N , M , D e una matrice dei tempi di processamento T), ritorni in output una formula proposizionale in formato DIMACS i cui modelli rappresentino soluzioni dell'istanza data del problema.