

АНАЛИЗ НА РЕШЕНИЕТО НА ЗАДАЧА ПРОГРАМИ

Можем да приемем, че програма в сектор с номер 1 започва да се предпроцесва в момент $t=0$. Нека с $tabeg_i$ означим началото на предпроцесване на програма в сектор с номер i . За начало имаме $tabeg_1=0$. Нека сме изчислили $tabeg_i$. Тогава $tabeg_{i+1}$ ще се изчисли, изхождайки от следните съображения:

$tabeg_{i+1} \geq tabeg_i + a_i$ – процесор A трябва да е завършил предпроцесването на програма в сектор с номер i , за да може да започне предпроцесването на програма в сектор $i+1$;

$tabeg_{i+1} + a_{i+1} \geq tabeg_i + a_i + b_i$ – процесор A трябва да завърши предпроцесването на програма в сектор с номер $i+1$ не по-рано от момента, в който процесор B ще се освободи от същинското изпълнение на програма в сектор с номер i . Това е необходимо, тъй като веднага след предпроцесването на програма в сектор с номер $i+1$ процесор B трябва да започне същинското и изпълнение. Тези две неравенства ни позволяват да напишем следната формула:

$$tabeg_{i+1} = tabeg_i + \max(a_i, a_i + b_i - a_{i+1}).$$

Тогава можем да напишем следната верига от равенства:

$$tabeg_n = tabeg_{n-1} + \max(a_{n-1}, a_{n-1} + b_{n-1} - a_n) = tabeg_{n-2} + \max(a_{n-2}, a_{n-2} + b_{n-2} - a_{n-1}) + \max(a_{n-1}, a_{n-1} + b_{n-1} - a_n) = \dots = tabeg_1 + \max(a_1, a_1 + b_1 - a_2) + \dots + \max(a_{n-2}, a_{n-2} + b_{n-2} - a_{n-1}) + \max(a_{n-1}, a_{n-1} + b_{n-1} - a_n) = \max(a_1, a_1 + b_1 - a_2) + \dots + \max(a_{n-2}, a_{n-2} + b_{n-2} - a_{n-1}) + \max(a_{n-1}, a_{n-1} + b_{n-1} - a_n).$$

Това е моментът, в който започва предпроцесването на програмата в последния сектор с номер n . Общото време на изпълнение ще бъде равно на

$$\max(a_1, a_1 + b_1 - a_2) + \dots + \max(a_{n-2}, a_{n-2} + b_{n-2} - a_{n-1}) + \max(a_{n-1}, a_{n-1} + b_{n-1} - a_n) + a_n + b_n.$$

При първото изпълнение на програмите по тази формула може да се сметне и запомни общото време за изпълнение. При следващите размени и изпълнения трябва да се забележи, че размяната на програми в секторите с номера i и j влияе максимум на четири от събираемите и, за да пресметнем новото време на изпълнение след размяната, не трябва да смятаме отново всичко, а трябва само да извадим от изчислената сума старите стойности на членовете на сумата, които се влияят от размяната, да направим размяната и да добавим новите стойности на същите членове.

Сложността на алгоритъма е $O(N+K)$.

Автор: Руско Шиков