

$$X = a * 7 + (a * 5 + 2)$$

⊕ 1: $a = 2$

2: $v = a * 7$

3: $u = a * 5$

4: $T = u + 2$

5: $X = v + T$

لكل node من دي

بتعمل عليه واهو فقط

مثل جمع - طرح - ضرب وقسمة

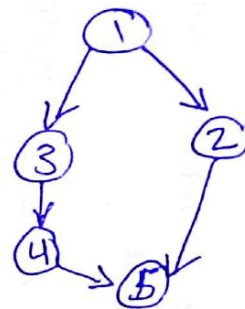
سه هتتفع تتعمل عليها مرة

واحدة زي $a * 5 + 2$

* السهم بيخرج من node الى

node التالية اذا كانت
~~node~~ قيمة node الاولى دخلت
في العملية التالية من node التالية

مثل السهم \rightarrow خرج من 1 الى 2
لكن \rightarrow قيمة 2 محتاجين
في 2
وهكذا



اهم حاجة في توزيع node على

Processor لازم يكون node

اللي هتتفد لا تعتمد على حاجة
أو تعتمد بس الحاجة اللي بتعتمد
عليها تكون اتنفذت

	P_1	P_2
1	1	
2	2	3
3	3	4
4	5	

مثال في أول مرة تفذت 1 بس عليه ؟؟

لوجيكا 3 وعابر اتفد لازم يكون 1 نهلت
لذلك تعتمد أو عايزة قيمة من 1

elementary graph Algorithms

- ① BFS (Breadth first Search)
- ② DFS (Depth first Search)
- ③ Task Duplication Based Algorithm.
- ④ Clustering Heuristic Algorithms.

في قوانين عايزه نفهم دلوقت

$$\text{Speed up} = \frac{\text{sequential execution}}{\text{Parallel scheduling}}$$

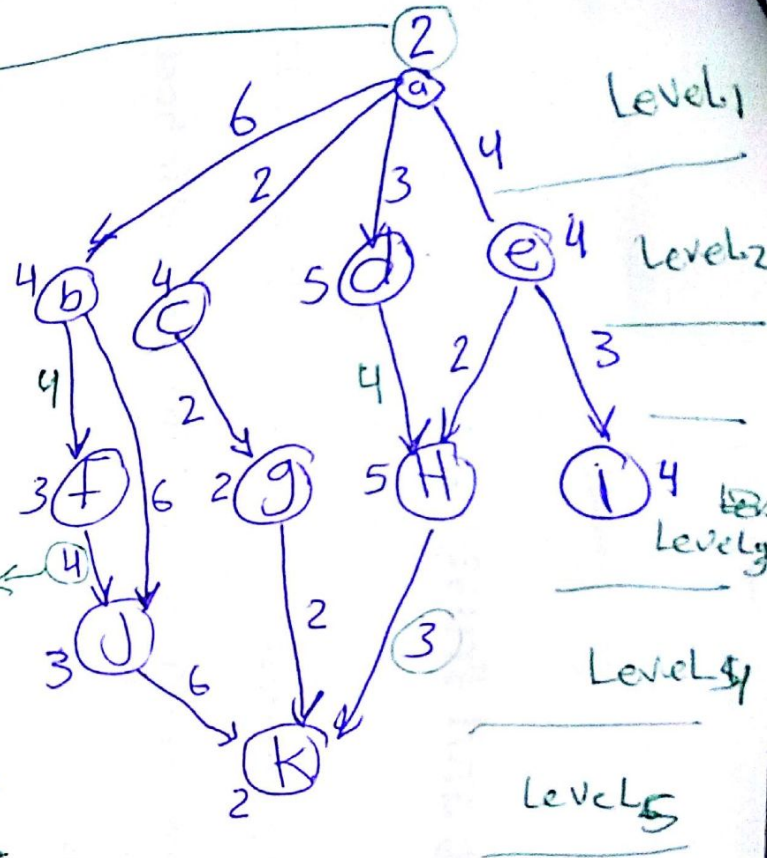
$$\text{efficiency} = \frac{\text{speed up}}{\text{no. processor}}$$

نقسم الرسالة إلى levels حسب BFS \Rightarrow

dePendency

الرقم الذي موجود
على كل دائرة دي

هو زمن تنفيذ Task
دي في Processor



الرقم الذي على كل
زمن delay لا تقال لتنفيذ

Task التالية من
مثال طما بخلص تنفيذ F وينتقل
إلى (j) هيا فتره وقت مقداره
4 فترة زمنية

نقسم الرسالة إلى levels حسب dePendency

level 1 \Rightarrow a

level 2 \Rightarrow b, c, d, e

level 3 \Rightarrow f, g, h, i

level 4 \Rightarrow j

Level 5 \Rightarrow k

$$\text{Speed up} = \frac{\text{sequential execution}}{\text{parallel scheduling}}$$

الوقت المجموع زمن التنفيذ لكل task موجودة عند
 الزمن الكلي الذي اتا وصلت له عند طريقة الرسالة التي فاتت

$$= \frac{2+4+4+5+4+3+2+5+4+3+2}{22}$$

$$= 1.72$$

$$E = \frac{\text{Speed up}}{\text{no of p}} = \frac{1.72}{3} \times 100 = 57\%$$

DFS \Rightarrow critical paths بحسب طريقة اسفل

هو أطول Path موجودة في الرسالة حسابها زاي
 زمن التغير + زمن delay

① a-b-f-j-k $2+6+4+4+3+4+6+2=34$

② a-b-j-k $2+6+4+3+6+2=29$

③ a-c-g-k $2+2+4+2+2+2+2=16$

④ a-d-h-k $2+3+5+4+5+3+2=24$

⑤ a-e-h-k $2+4+4+2+5+3+2=22$

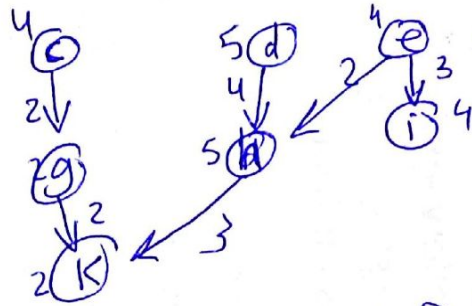
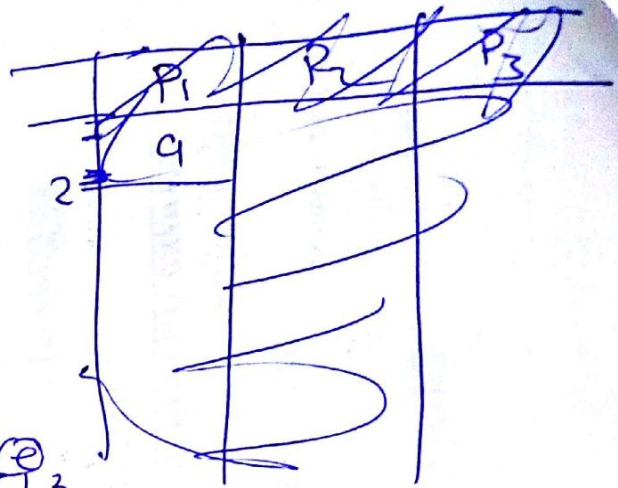
⑥ a-e-l $2+4+4+3+4=17$

Path هو أطول ~~في~~ ~~الرسالة~~ ~~التي~~ ~~فاتت~~

الخطوة الثانية

حذف Path ده من
الرسمة

فتبقى الرسمة على كده



واغير حساب Path كان

C-g-k	$4+2+2+2+2=12$	
② d-h-k	$5+4+5+3+2=19$	
e-h-k	$4+2+5+3+2=16$	
e-l	$4+3+4=11$	

أطوره ~~d-h-k~~ ط ايجي أنفذ
هلاقى
dependencies محتايين d, h

④ اغير الحسابات مرة آخر بعد حذف

C-g-k	$4+2+2+2+2=12$
h-k	$5+3+2=10$
e-h-k	$4+2+5+3+2=16$
e-l	$4+3+4=11$

ط ايجي أنفذ e-h-k هلاقى
عابزة dependent

أغير الحسابات مرة أخرى بعد حذف



C-g-K

e-l

$$4+2+2+2+2 = 12$$

$$4+3+4 = 11$$

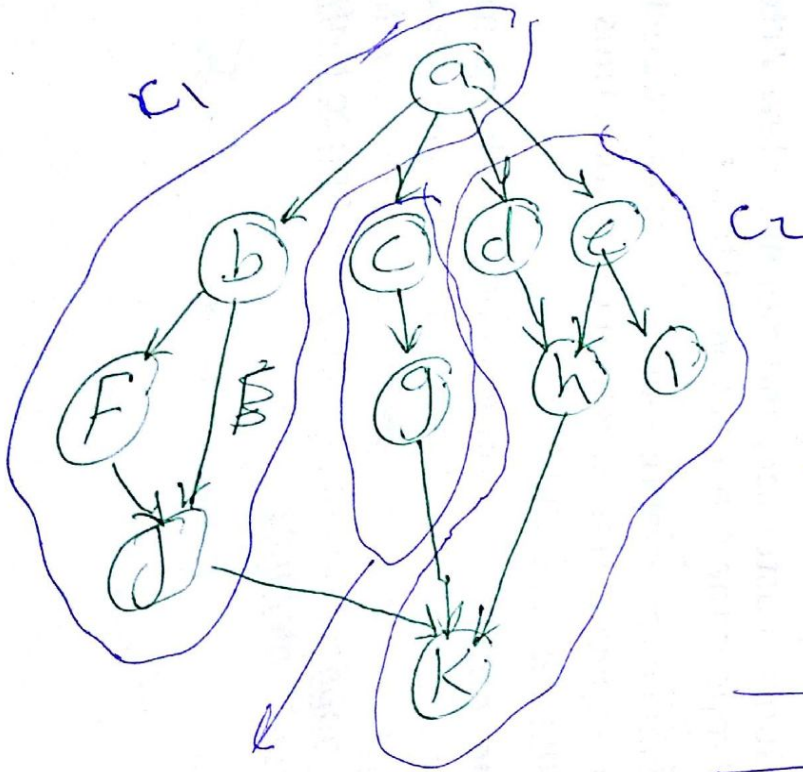
هنگام C-g اول در دسترس است.

التوزيع لجزى Processor
اعراف
ازاي

	P ₁	P ₂	P ₃
1	a		
2			
3			
4	c		
5			
6			
7	g		d
8			
9		b	
10			
11			
12			e
13		f	
14			
15		j	i
16			
17			
18			

Clustering

يعمل كذا cluster و كل cluster في
تنفيذ P و P و P



الهدف من التوزيع هو
الوصول الى أقل وقت
يتحقق في Try و error
تجرب كثير

Speedup = $\frac{38}{25} = 1.52$

$E = 50.6\%$

	P ₁	P ₂	P ₃
1	a		
2			
3	b	c	
4			d
5	f		
6		g	
7	j		e
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			
21			
22			
23			
24			
25			