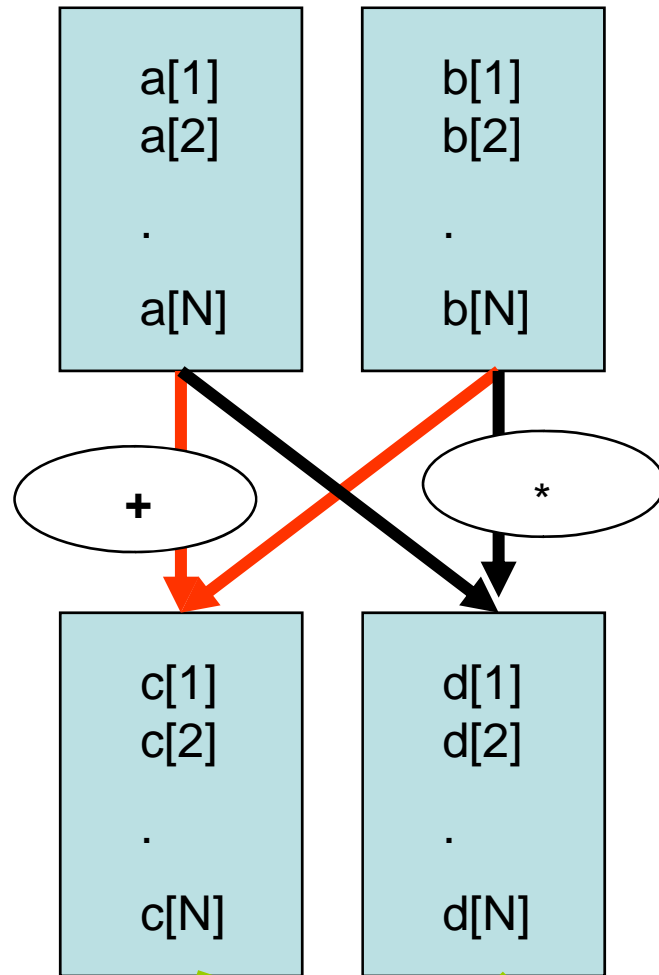


¿En qué se puede cooperar? (extracción del paralelismo implícito)

¿Es posible cooperar de distintas formas? (niveles de paralelismo)

¿Hasta qué punto se puede cooperar? (granularidad)



Hay que sumar dos arrays, A y B, por un lado y multiplicar sus componentes por otro lado. Se comparan componente a componente los dos arrays resultantes y se suman los componentes menores.

S es la suma de los componentes menores de los vectores c y d

Formas de describir el problema

S=0

```
for i=1 to N do
  begin
    c[i]=a[i]+b[i];
  end;
```

```
for i=1 to N do
  begin
    d[i]=a[i]*b[i];
  end;
```

```
for i=1 to N do
  begin
    if (c[i]<d[i]) then S=S+c[i] else S=S+d[i];
  end;
```

Tres bucles diferentes

S=0

```
for i=1 to N do
  begin
    c[i]=a[i]+b[i];
    d[i]=a[i]*b[i];
    if (c[i]<d[i]) then S=S+c[i] else S=S+d[i];
  end;
```

**Un único bucle en el
que se trabaja con los
componentes de los
array en las distintas
iteraciones**

S=0

```
for i=1 to N do
```

```
begin
```

```
    c[i]=a[i]+b[i];
```

```
end;
```

```
for i=1 to N do
```

```
begin
```

```
    d[i]=a[i]*b[i];
```

```
end;
```

Tres bucles diferentes

```
for i=1 to N do
```

```
begin
```

```
    if (c[i]<d[i]) then S=S+c[i] else S=S+d[i];
```

```
end;
```

Este bucle tiene dependencias de tipo
RAW con los dos bucles anteriores
(necesita leer C y D que son generados
por esos bucles)

Las tareas
correspondientes a
cada uno de estos
bucles son
independientes entre sí
y se pueden ejecutar
en paralelo

Se puede considerar una tarea previa que carga en el array C los componentes menores, pero desde el punto de vista de las tareas que se ejecutan en paralelo no cambiaría mucho

```
for i=1 to N do
```

```
  begin
```

```
    c[i]=a[i]+b[i];
```

```
  end;
```

```
for i=1 to N do
```

```
  begin
```

```
    d[i]=a[i]*b[i];
```

```
  end;
```

```
for i=1 to N do
```

```
  begin
```

```
    if (d[i]<c[i]) then c[i]=d[i];
```

```
  end;
```

```
for i=1 to N do
```

```
  begin
```

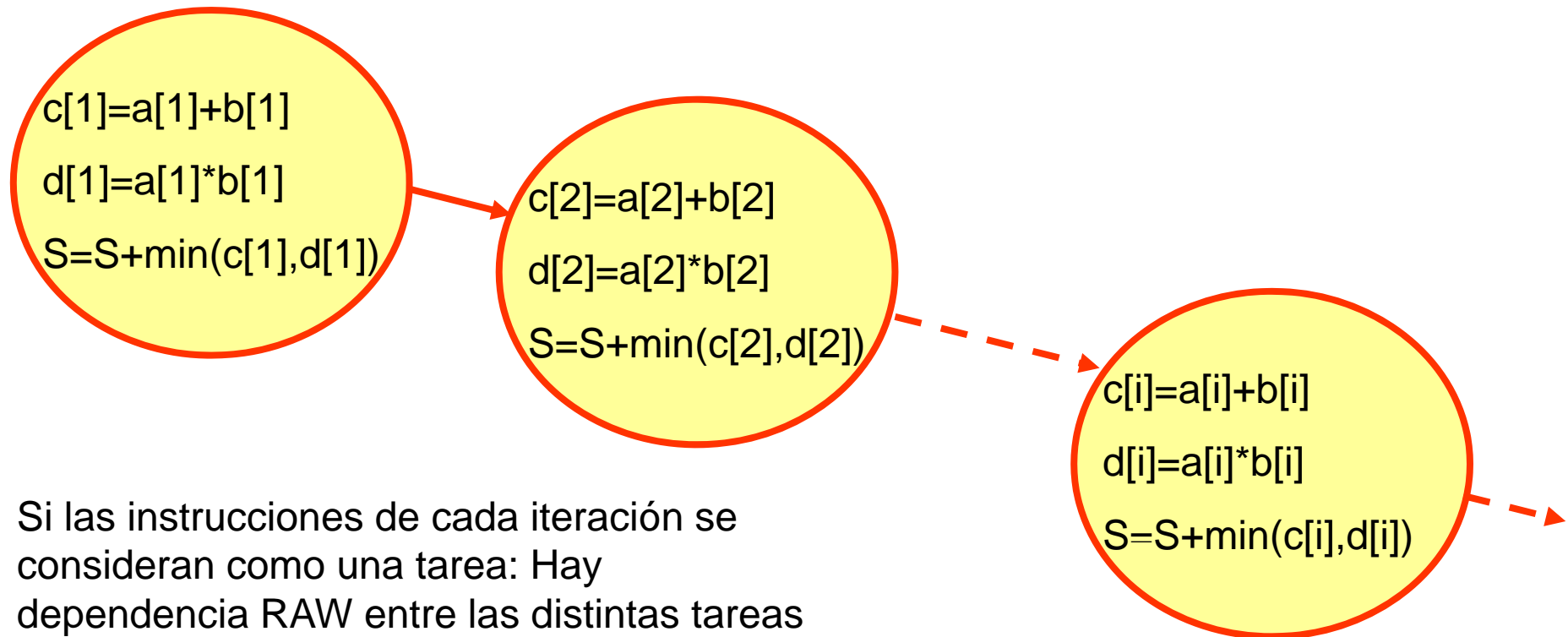
```
    S=S+c[i];
```

```
  end;
```

S=0

```
for i=1 to N do
  begin
    c[i]=a[i]+b[i];
    d[i]=a[i]*b[i];
    if (c[i]<d[i]) then S=S+c[i] else S=S+d[i];
  end;
```

En la descripción de un único bucle



S=0

```
for i=1 to N do
  begin
    c[i]=a[i]+b[i];
    d[i]=a[i]*b[i];
    if (c[i]<d[i]) then S=S+c[i] else S=S+d[i];
  end;
```

Si nos fijamos en las dependencias de las operaciones que hay que realizar en las distintas iteraciones:

Las N sumas de los elementos de los arrays a[] y b[] son independientes

Los N productos de los elementos de los arrays a[] y b[] son independientes

Las N comparaciones de los elementos de los arrays c[] y d[] son independientes (una vez que se tienen esos arrays calculados)

La obtención de S parece implicar cierto procesamiento secuencial

```

for i=1 to N do
  begin
    c[i]=a[i]+b[i];
  end;

```

```

for i=1 to N do
  begin
    if (d[i]<c[i]) then c[i]=d[i];
  end;

```

```

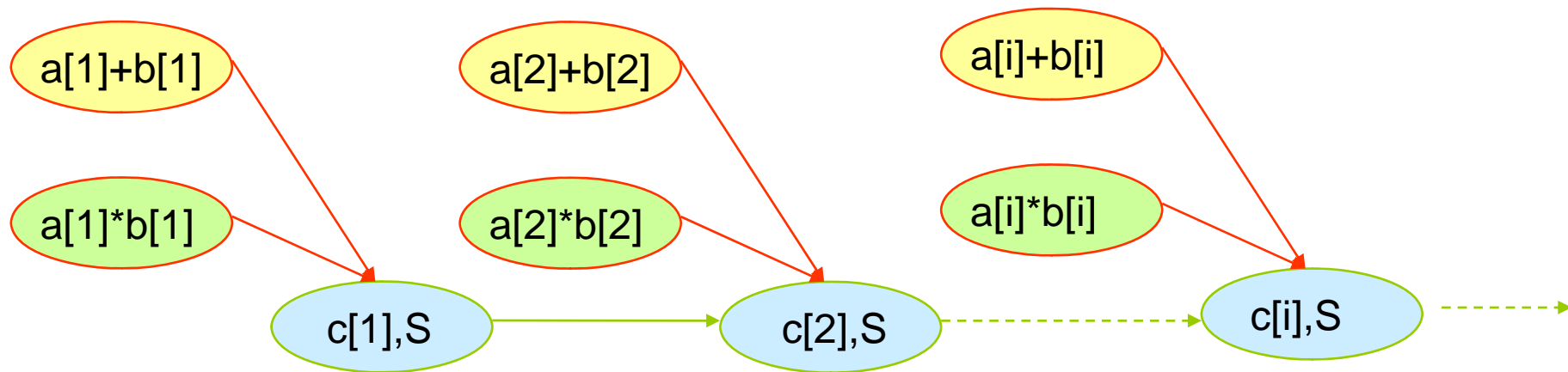
for i=1 to N do
  begin
    d[i]=a[i]*b[i];
  end;

```

```

for i=1 to N do
  begin
    S=S+c[i];
  end;

```



Las sumas y los productos de las componentes de A y B son independientes

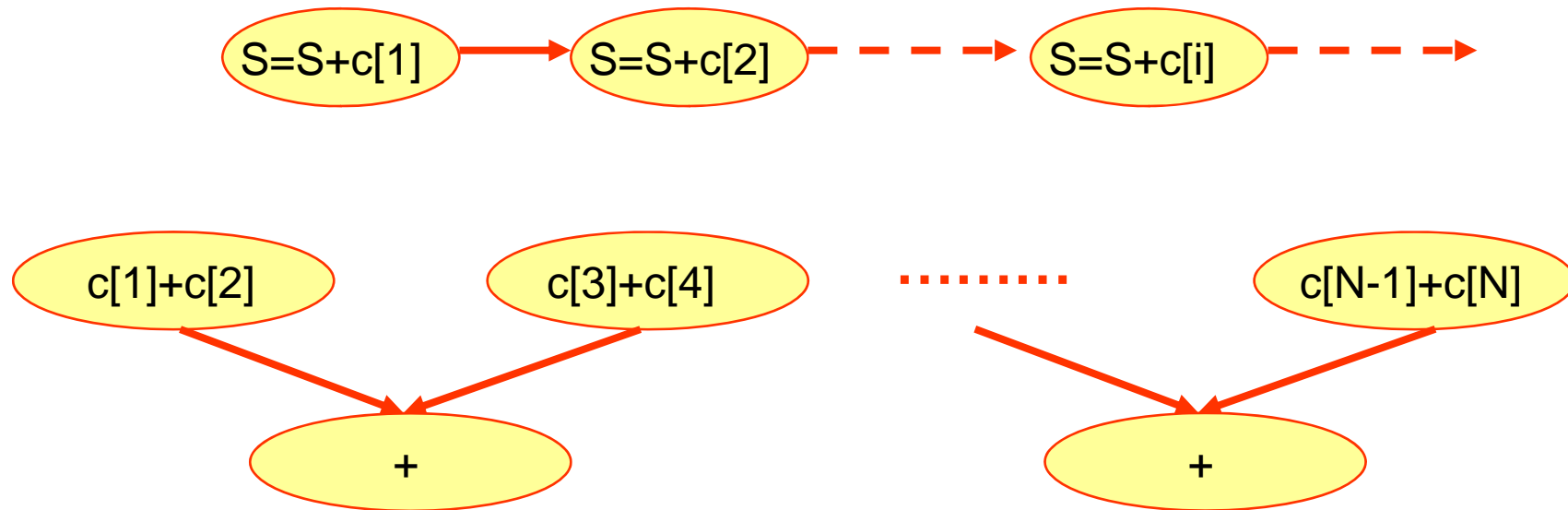

```
for i=1 to N do
```

```
  begin
```

```
     $S = S + c[i];$ 
```

```
  end;
```

Habría N pasos
secuenciales de sumas



Pero también se podrían calcular $N/2$ sumas en paralelo (suponiendo N par), cuyos resultados también se van sumando por parejas, para tener al final un número de pasos de suma del orden de $\log_2(N)$