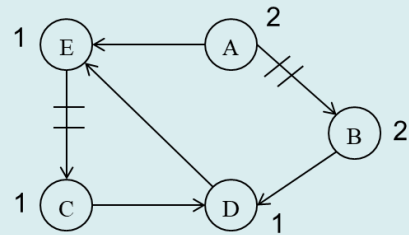


Parallelization 答疑

```
for (i = 2; i < n; i++) {
    A[i] = E[i] * 2;
    B[i] = A[i-2] * A[i-2];
    C[i] = E[i-2] + 5;
    D[i] = B[i] + C[i];
    E[i] = D[i] + 9;
}
```

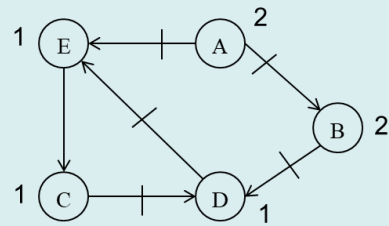


1) 关于 A[i] 与 E[i] 之间存在数据依赖的问题。

首先，数据 A[i] 计算需要先读 E[i]，所以两条指令之间存在依赖关系。因此，计算 A[i] 的指令必须先于 E[i] 执行。

2) 关于最大并行化问题

```
for (i = 2; i < n; i++) {
    A[i] = E[i] * 2;
    B[i+1] = A[i-1] * A[i-1];
    E[i-1] = D[i-1] + 9;
    C[i+1] = E[i-1] + 5;
    D[i] = B[i] + C[i];
}
```



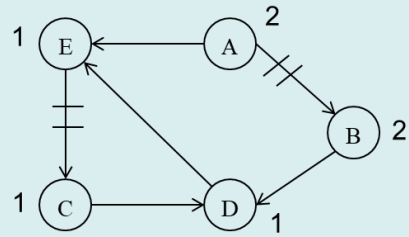
p0		A2	A3
p1	B2	B3	B4
p2	C2	E1 C3	E2 C4
p3		D2	D3

◆ Iteration period = 2

是可以实现 iteration period=2 的。

3) 关于数据流图的问题。

```
for (i = 2; i < n; i++) {  
    A[i] = E[i] * 2;  
    B[i] = A[i-2] * A[i-2];  
    C[i] = E[i-2] + 5;  
    D[i] = B[i] + C[i];  
    E[i] = D[i] + 9;  
}
```



右边的指令间的数据流图，原来课件上是错误的，因此导致了问题。感谢课堂上提出疑问的同学们，我后面发现是因为数据流图画错的问题。

A 与 E 之间的数据依赖应该是 E 必须等到 A 计算好再计算的，因此是 A 向 E 的边。