آزمون نرم افزار – بخش۲–۷

نمایش جبری گراف‌ها

صدیقه خوشنویس
دانشگاه آزاد اسلامی – واحد شهرقدس

گراف و نمایش جبری آن

- گاهی نیاز داریم با نوشتن ابزار یا **script**های آزمون، آزمایش را خودکار کنیم.
- برای نوشتن ابزار، نیاز به ساختمان داده استانداردی برای گراف داریم.
- نمایش جبری گراف (عبارت منظم) امکان انجام عملیات مفید ریاضی را به ما می دهد.

- به هر یال یک برچسب بزنید (می تواند نشان دهنده معنای خاص و یا دلخواه باشد)
- عملیات روی یالها:
  - اتصال (**concatenation**) (ضرب):
    - اگر پس از یال **a**، یال **b** واقع شود، **a** و **b** متصل هستند و داریم: **a*b** یا **ab**
  - انتخاب (جمع):
    - اگر بتوانیم یال **a** یا یال **b** را ادامه دهیم، **a** و **b** جمع می شوند و داریم: **a+b**

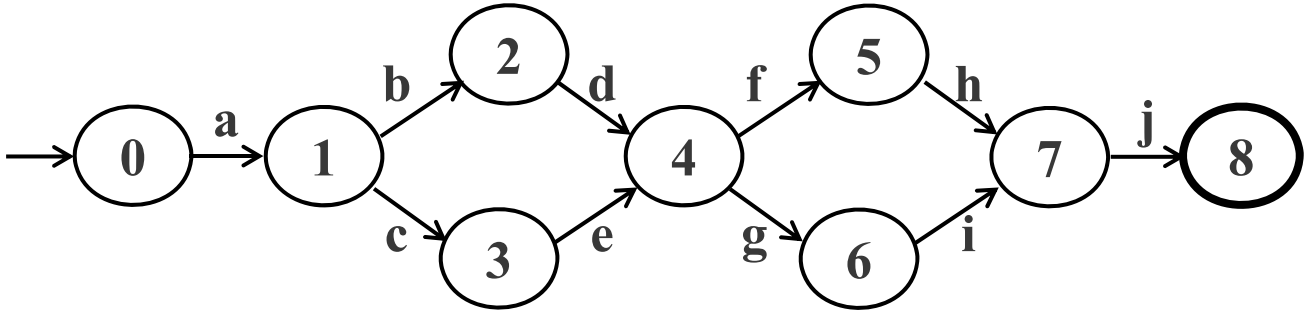
گراف و نمایش جبری آن...

- **حاصلضرب مسیر (Path Product):**
  - دنباله ای از یالها که در هم ضرب شده اند (متصل شده اند)

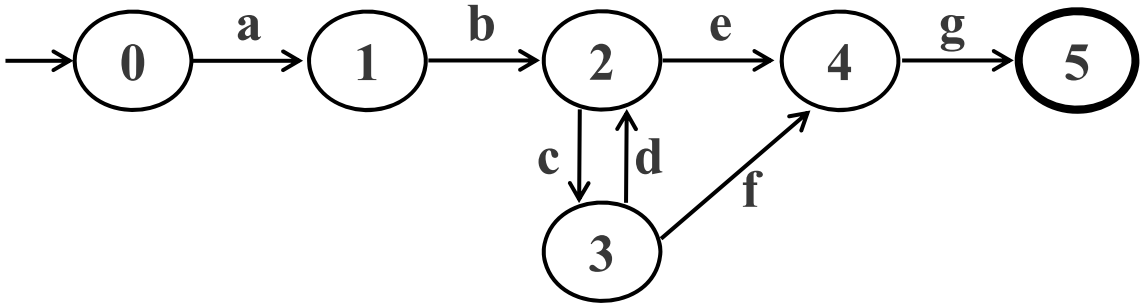
- **عبارت مسیر (Path Expression):**
  - یک حاصلضرب مسیر؛ یا چند حاصلضرب مسیر که با هم جمع شده اند

- **حلقه (loop):**
  - حلقه یک نوع ضرب متوالی یک یال (یا یک عبارت مسیر) در خودش است.
  - آن را به صورت توان نشان می دهیم

مثال



Path Expression = abdfhj + abdgij + acegij + acefhj



Path Products : A = abeg, B = (cd)*, C = cf

Path Expression = ab (cd)* (e + cf) g

سری به ریاضی بزنیم!

- **حاصلضرب مسیر (Path Product):**
  - جابجایی پذیر نیست؛ زیرا: $AB \mathrel{!=} BA$
  - شرکت پذیر هست؛ زیرا: $A(BC) = (AB)C = ABC$
  - توزیع پذیر هست؛ زیرا: $A(B + C) = AB + AC$
  - اگر A در خودش ضرب شود: حلقه (اسلاید بعد)

- **حاصلجمع مسیر (Path Summation):**
  - جابجایی پذیر هست؛ زیرا: $A+B = B+A$
  - شرکت پذیر هست؛ زیرا: $A+(B+C) = (A+B)+C = A+B+C$
  - توزیع پذیر هست؛ زیرا: $(B + C)D = BD + CD$
  - اگر A با خودش جمع شود: قانون جذب: $A+A=A$

نمایش جبری حلقه ها در گراف

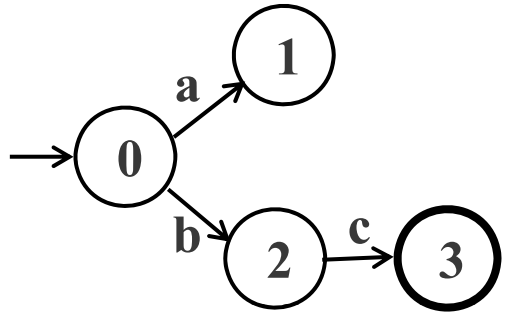
- حلقه به حداقل یک بار تکرار = $A^{+} = AA^{*}$
- محدودیت روی تعداد تکرار: $A^{\underline{n}} = A^{0} + A^{1} + \ldots + A^{n}$
  - مثلاً: $A^{\underline{3}} = A^{0} + A^{1} + A^{2} + A^{3}$
  - یعنی اصلا وارد حلقه نمی شویم (۰بار اجرا)؛ یا ۱ بار حلقه را اجرا می کنیم؛ یا ۲ بار و حداکثر ۳ بار.

- چند رابطه مهم در مورد حلقه ها

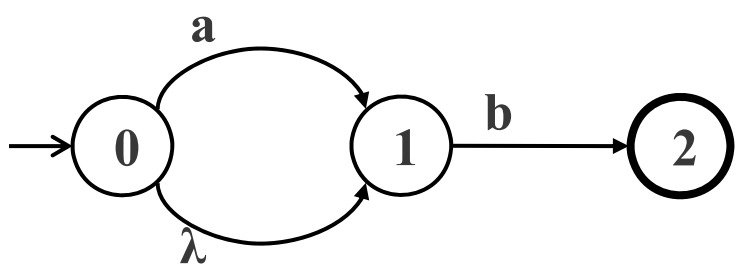
  - $A^{\underline{n}} + A^{\underline{m}} = A^{\underline{max(n,m)}}$
  - $A^{\underline{n}} A^{\underline{m}} = A^{\underline{n+m}}$
  - $A^{\underline{n}} A^{*} = A^{*} A^{\underline{n}} = A^{*}$
  - $A^{\underline{n}} A^{+} = A^{+} A^{\underline{n}} = A^{+}$
  - $A^{*} A^{+} = A^{+} A^{*} = A^{+}$

عملوندهای همانی

- عملوند همانی ضرب: λ
  - (یک مسیر خالی (بدون برچسب))(شکل پایین، بین ۰و۱)
- عملوند همانی جمع: Φ
  - (یک مسیر پوچ (که وجود ندارد))(شکل بالا، بین ۱و۳)



$pe = bc + a\Phi = bc$

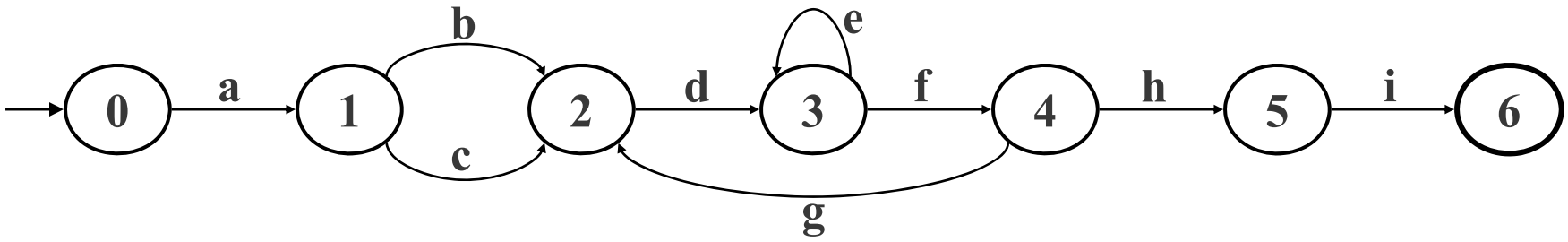


$pe = (a + \lambda)b = ab + \lambda b = ab + b$

- قوانین همانی ضرب:
  - $\lambda + \lambda = \lambda$
  - $\lambda A = A\lambda = A$
  - $\lambda^n = \lambda^{\underline{n}} = \lambda^* = \lambda^+ = \lambda$
  - $\lambda^+ + \lambda = \lambda^* = \lambda$
- قوانین همانی جمع:
  - $A + \Phi = \Phi + A = A$
  - $A\,\Phi = \Phi A = \Phi$
  - $\Phi^* = \lambda + \Phi + \Phi^2 + \ldots = \lambda$

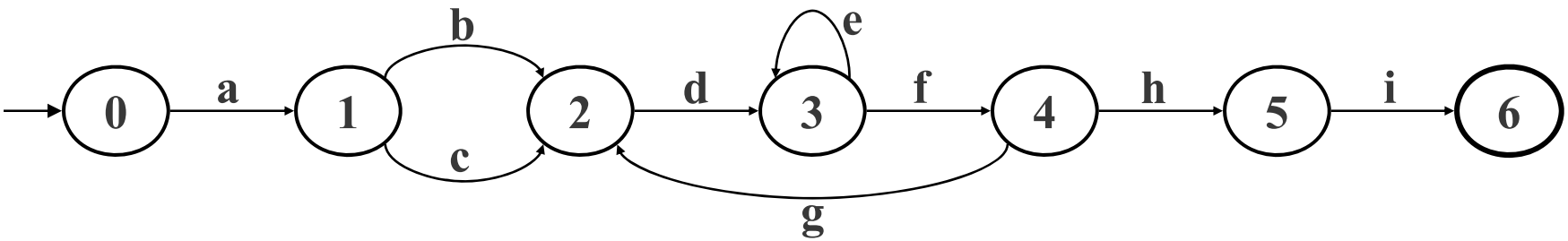
تبدیل گراف به عبارت مسیر (PE)

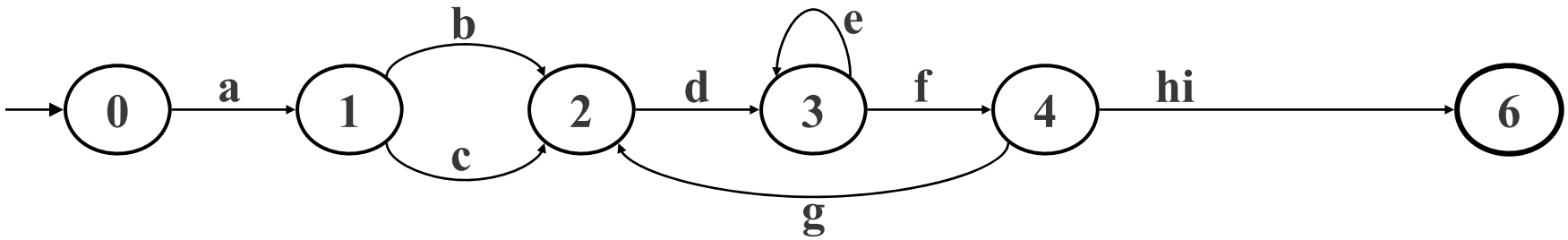
- این الگوریتم چهار گام دارد:
- برای توضیح این مراحل گراف زیر را در نظر بگیرید:



گام اول – یالهای متوالی

- همه یالهای متوالی را ترکیب کنید (یال های متوالی را در هم ضرب کنید).
- به طور دقیق تر: برای هر گره ای که تنها یک یال ورودی و یک یال خروجی دارد، گره را حذف کنید و دو یال را به هم بچسبانید و عبارت مسیر آنها را در هم ضرب کنید.
- مثال: در گراف زیر: یال های h و i با هم ترکیب می شوند.



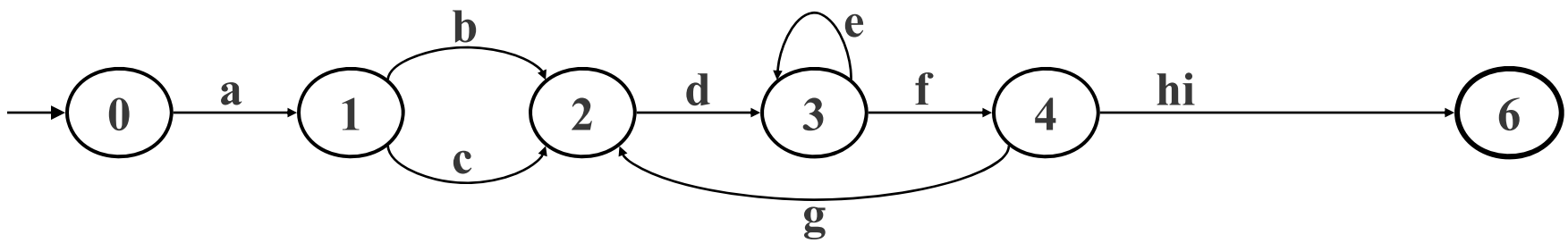


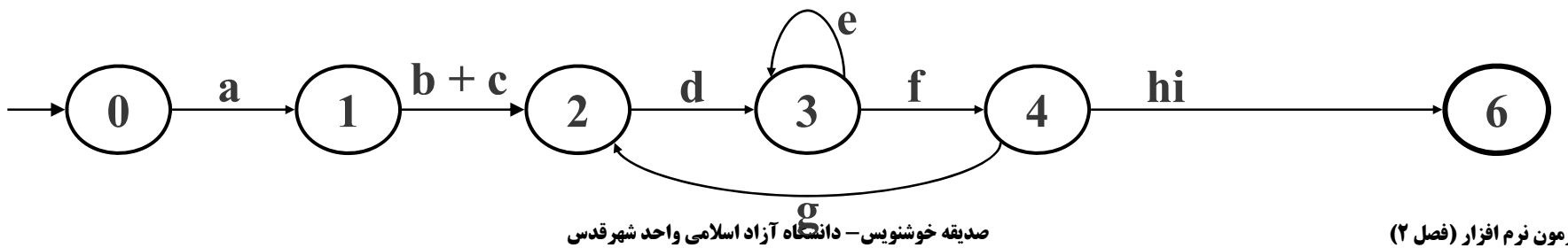
گام دوم – یالهای موازی

- همه یالهای موازی را ترکیب کنید (یال های موازی را با هم جمع کنید).

- به طور دقیق تر: برای هر زوج از یالها که گره مبدأ و مقصدشان یکسان است یالها را حذف کنید و به جای آنها یک یال ایجاد کنید که برچسب آن عبارت مسیر حاصل از جمع دو عبارت مسیر اولیه است کنید.

- مثال: یال های b و c با هم ترکیب می شوند:

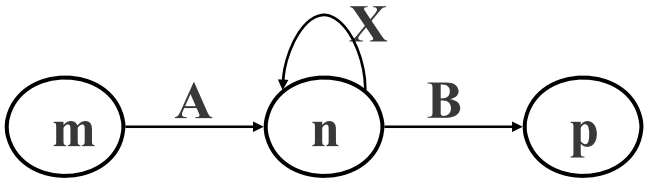


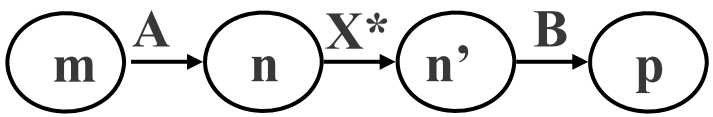


گام سوم – حلقه روی گره

- همه حلقه های روی یک گره را ترکیب کنید.
- یک گره ساختگی بعد از گره حلقه ایجاد کنید.
- یک یال از گره حلقه به گره ساختگی ایجاد کنید که عبارت مسیر آن، دارای نما باشد.
- سه یال خواهیم داشت: یال ورودی به گره حلقه، یال میان گره حلقه و گره ساختگی، یال خروجی از گره ساختگی؛ آنها را با هم ترکیب کنید (ضرب، مثل گام ۱)

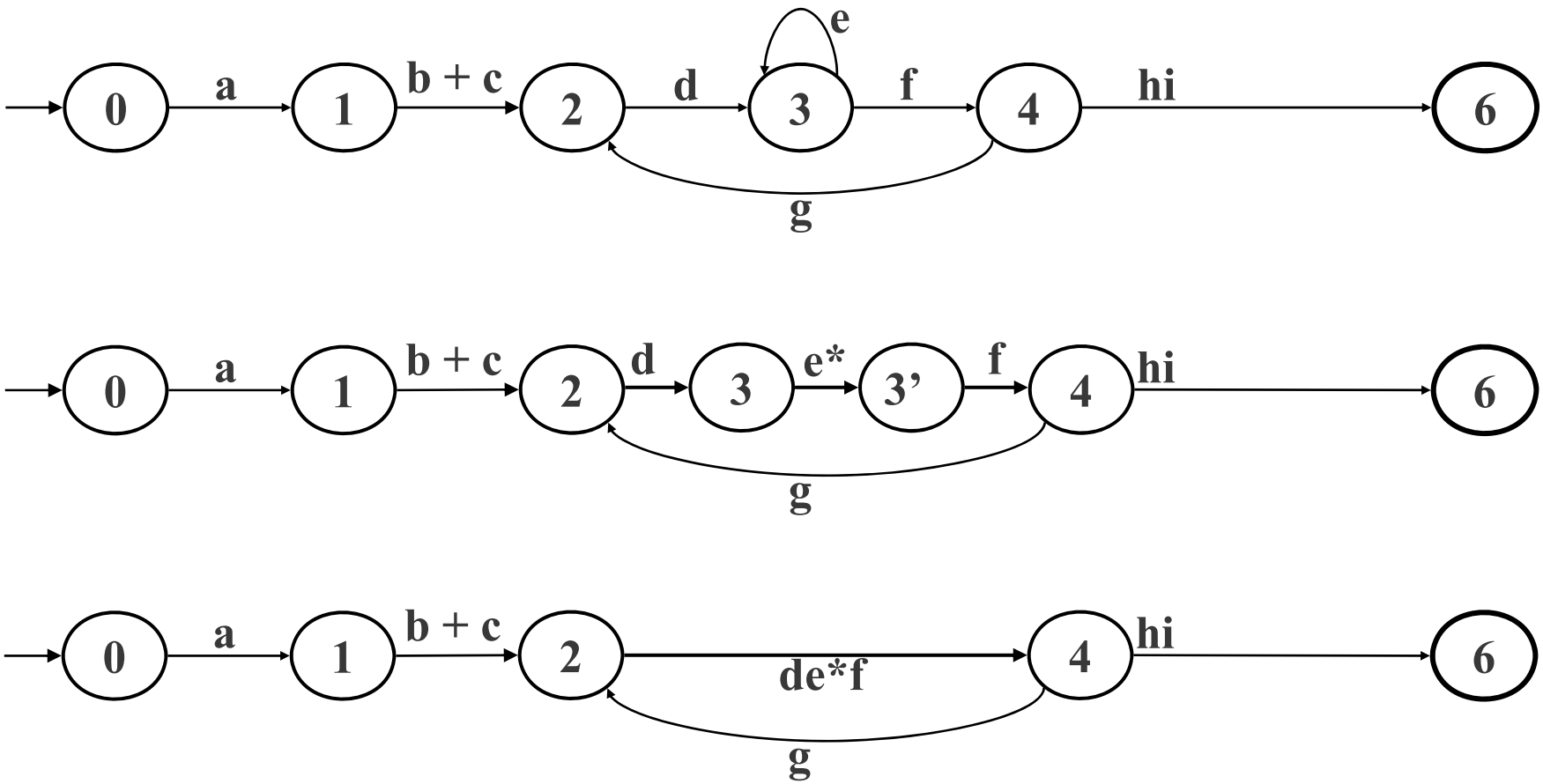
- به طور دقیق تر: برای هر گره **n** که دارای یال ورودی **A**، یال خروجی **B** و حلقه با برچسب **X** روی خودش است:
  - **X** را حذف کنید،
  - یک گره ساختگی **n'** بسازید.
  - یک یال از **n1** به **n'** با برچسب X^* ایجاد کنید. یال **B** از **n'** خارج می شود.
  - حالا گره های **n** و **n'** را حذف و سه یال را با هم ترکیب کنید = AX^*B





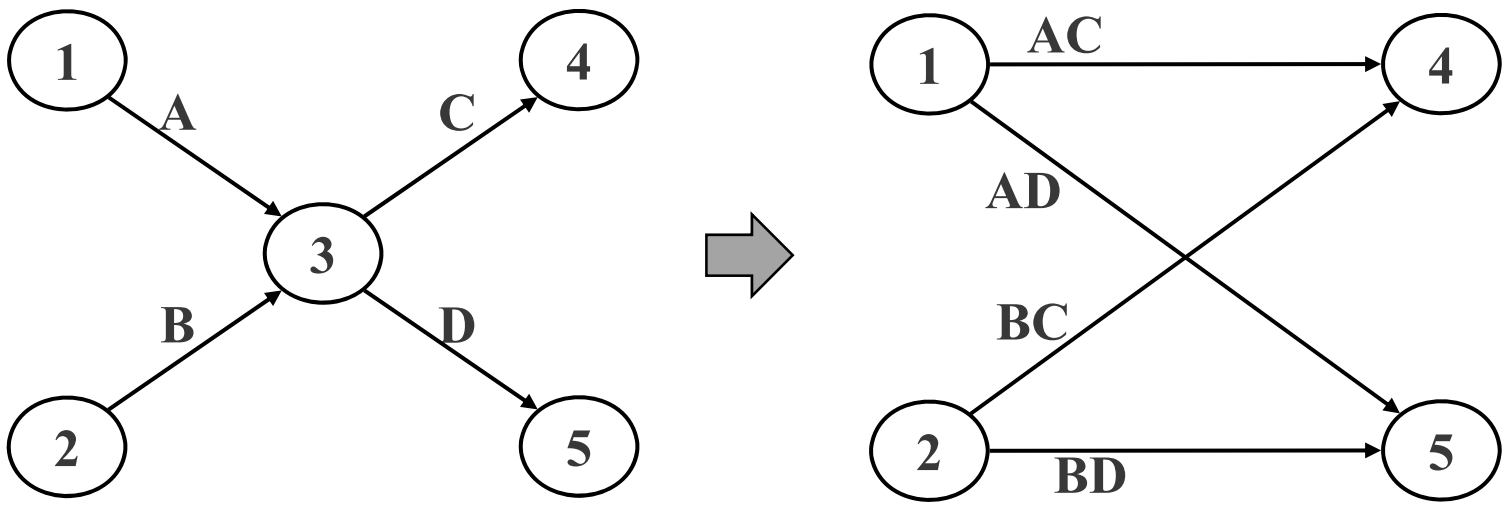
گام سوم – حلقه روی گره...

- مثال:



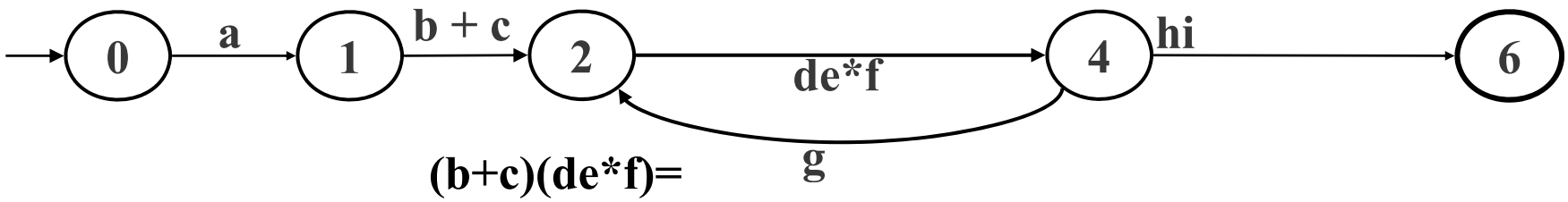
گام چهارم– حذف برخی گره ها به تشخیص آزمونگر

- برای موارد خاص که گام های ۱ تا ۳ قابل اجرا نباشد این گام را به کار می بریم.
- یک گره غیرآغازین و غیرپایانی را انتخاب کنید.
- آن را با یک یا چند یال از همه گره های قبلی به همه گره های بعدی جایگزین کنید.
- عبارت مسیر مربوط به همه یالهای ورودی و خروجی آن گره را در هم ضرب کنید.

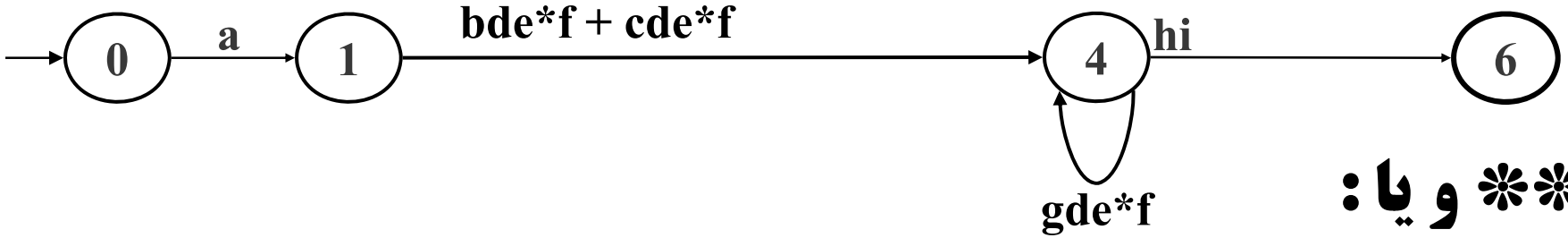


گام چهارم– حذف گره به تشخیص آزمونگر

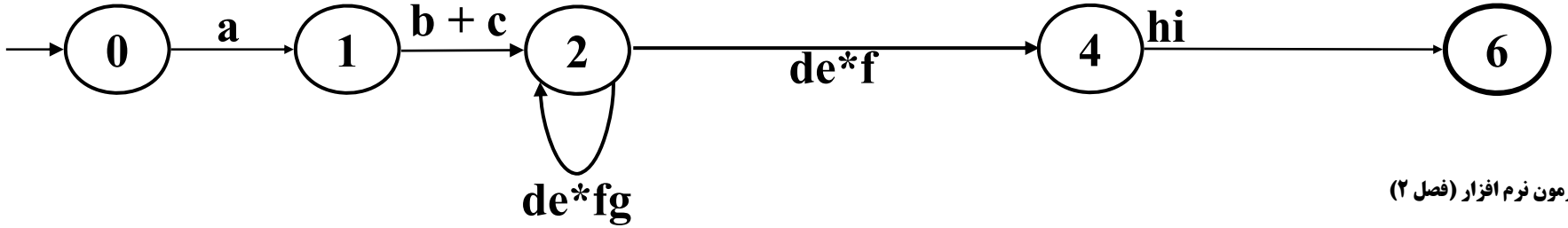
- برای مثال قبلی، گره 2 را حذف می کنیم
- یالهای (1,2) و (2,4) تبدیل به یک یال می شوند
- یالهای (4,2) و (2,4) تبدیل به یک دور روی گره می شوند



(b+c)(de*f)=



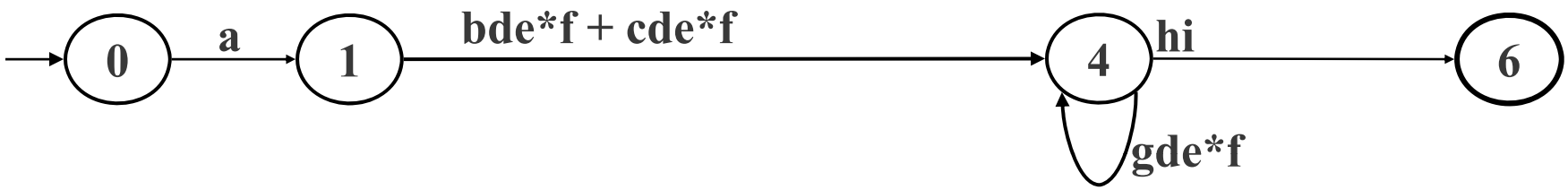
**** و یا :

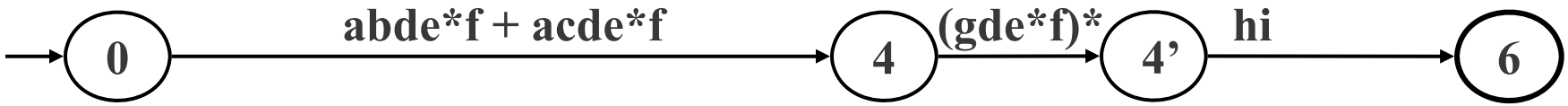
(

)

تکرار گام های اول تا چهارم

- گام های اول تا چهارم را آنقدر تکرار می کنیم تا فقط یک یال باقی بماند...

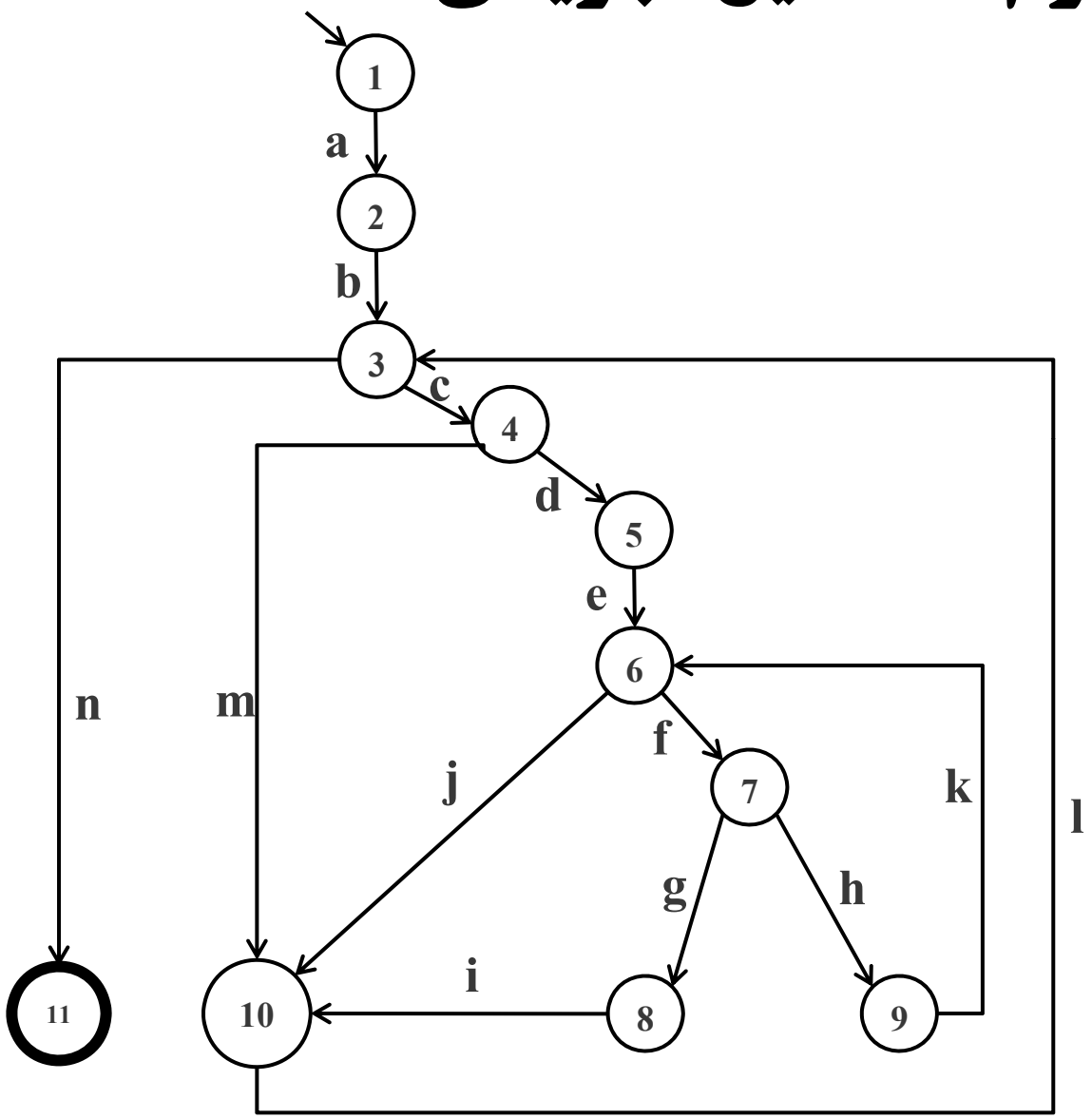




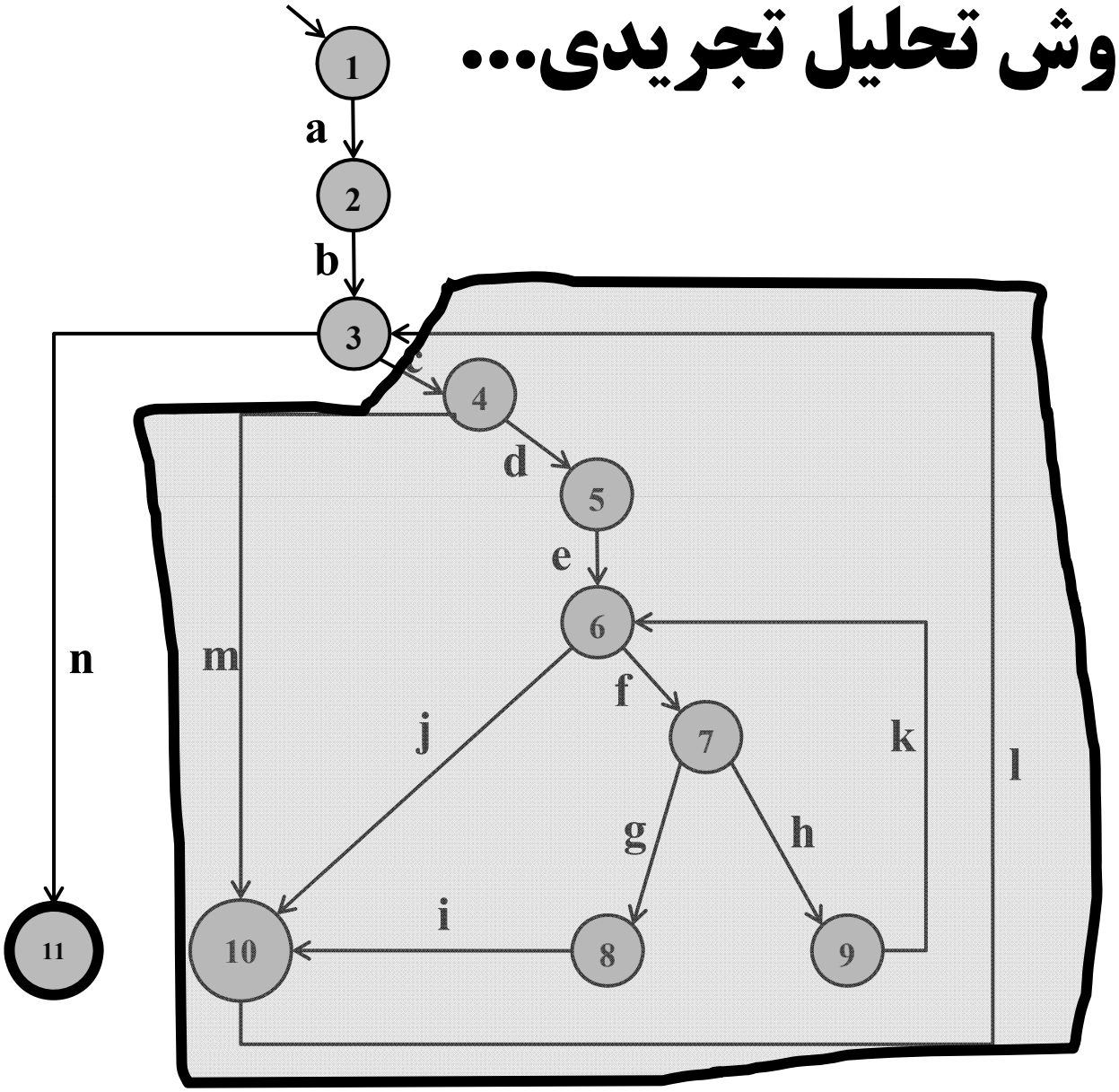
abde*f (gde*f)* hi + acde*f (gde*f)* hi

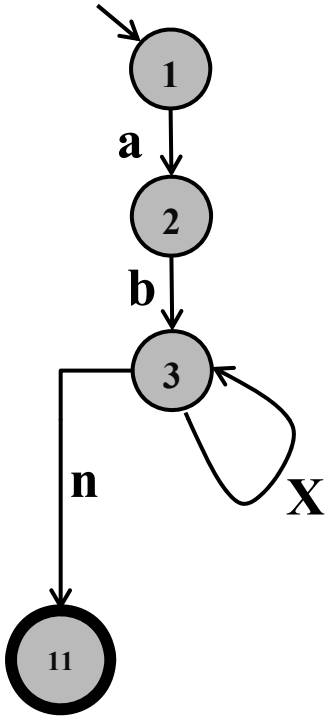
روش دوم – تحلیل تجریدی

- مثال



روش تحلیل تجریدی...

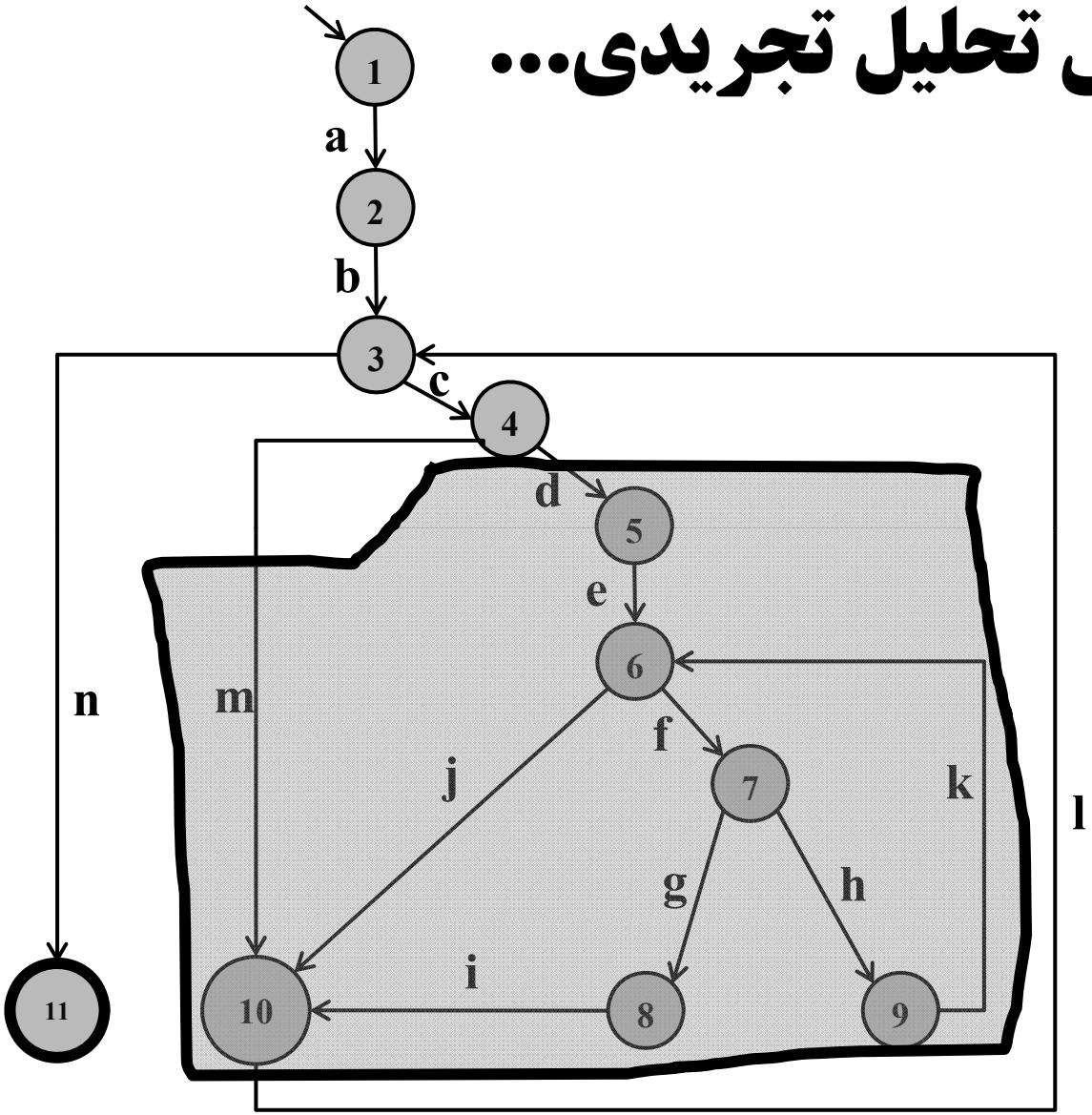




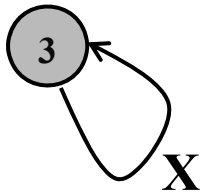
$\Rightarrow$P.E. = abX*n

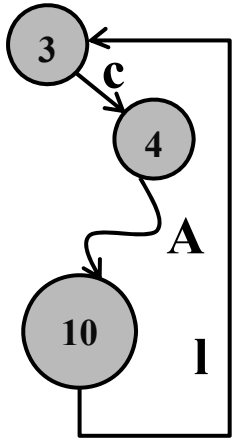
X=?

روش تحلیل تجریدی...



X=?

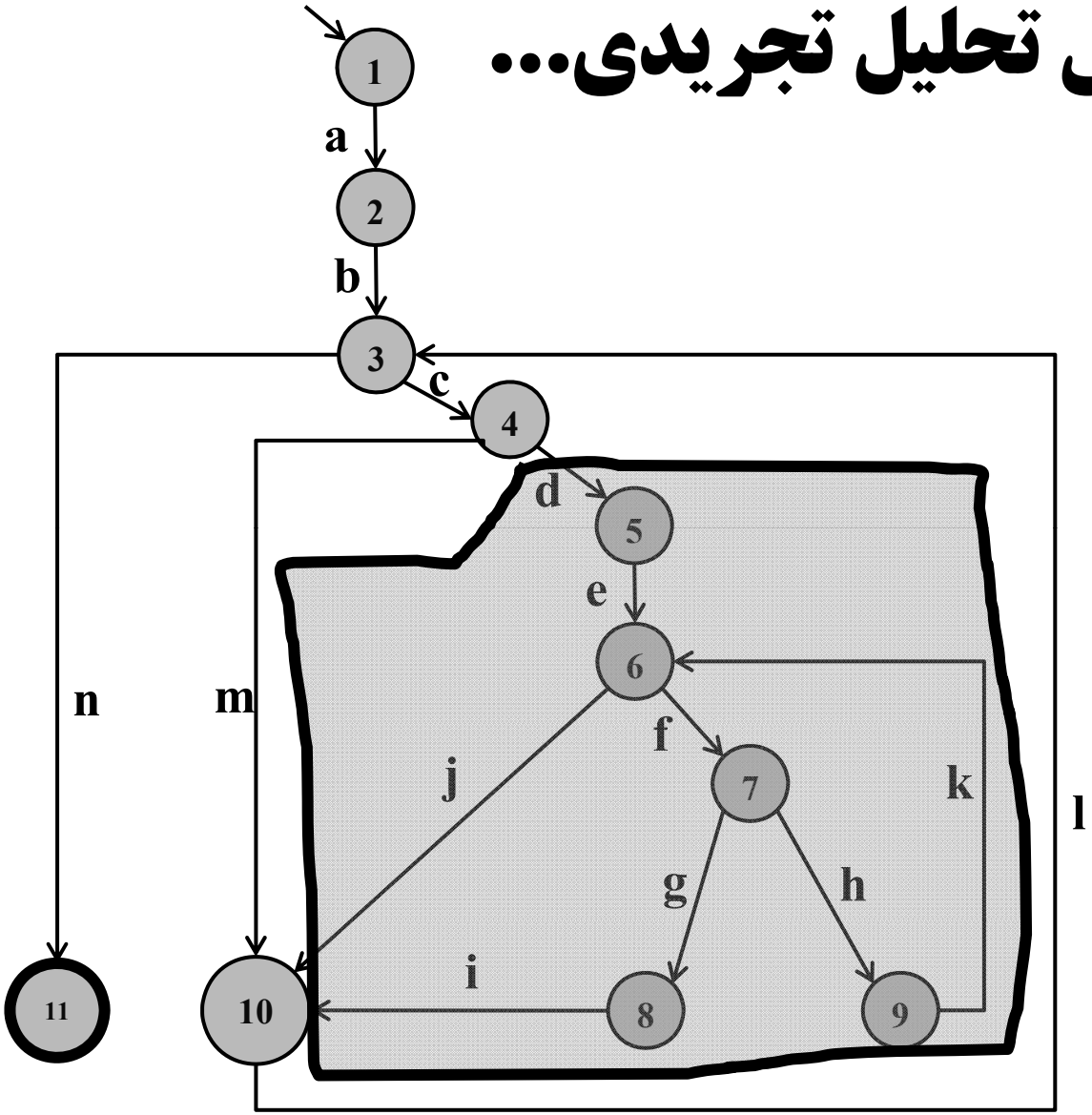




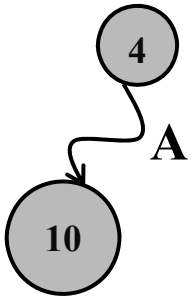
$\Rightarrow$X=cAl

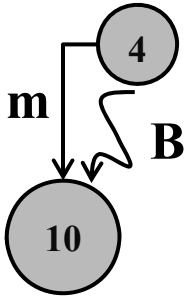
A=?

روش تحلیل تجریدی...



A=?

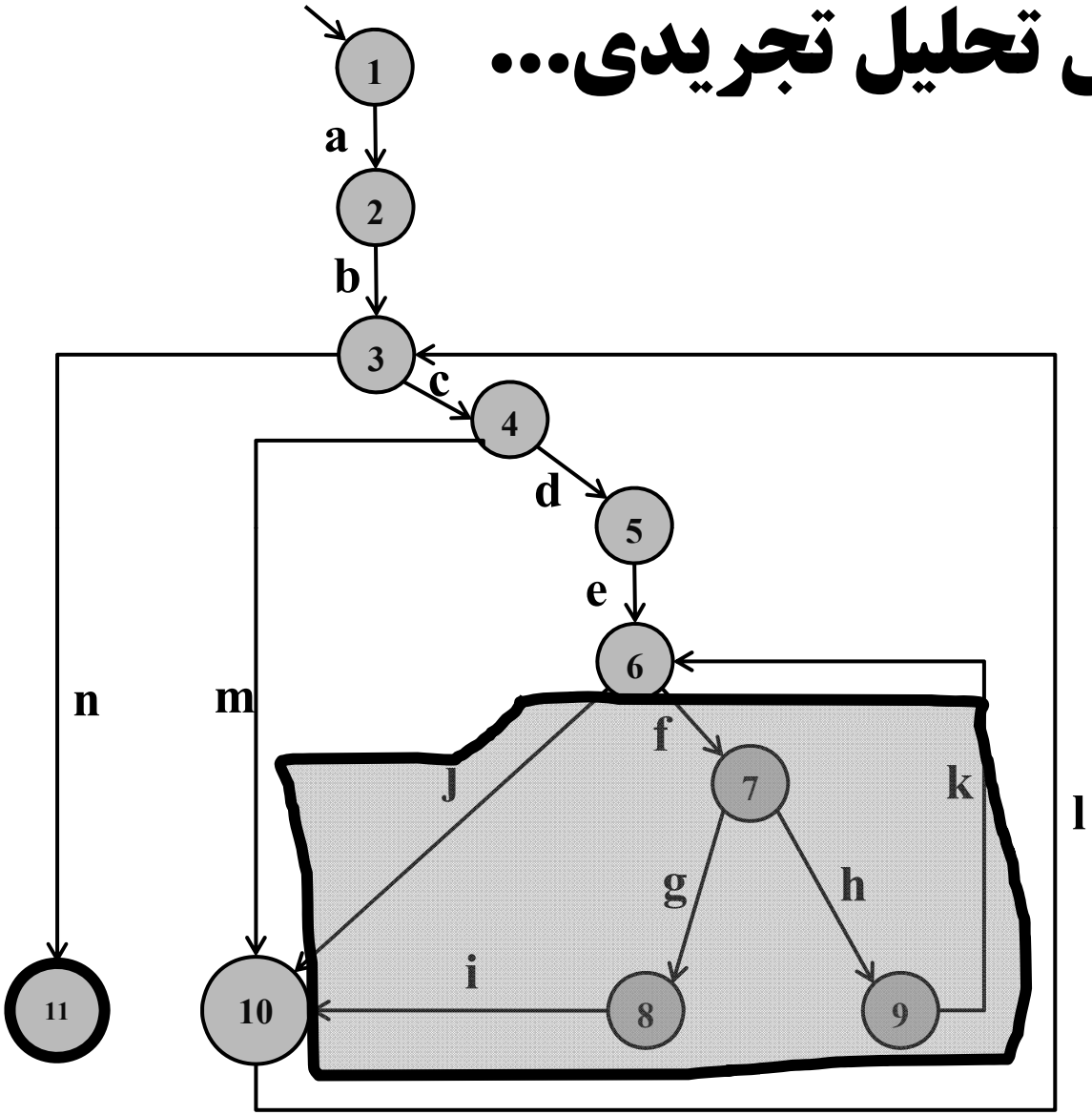


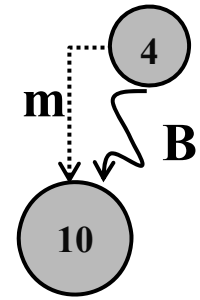


$\Rightarrow$A=m+B

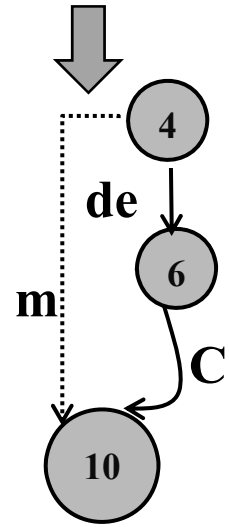
B=?

روش تحلیل تجریدی...





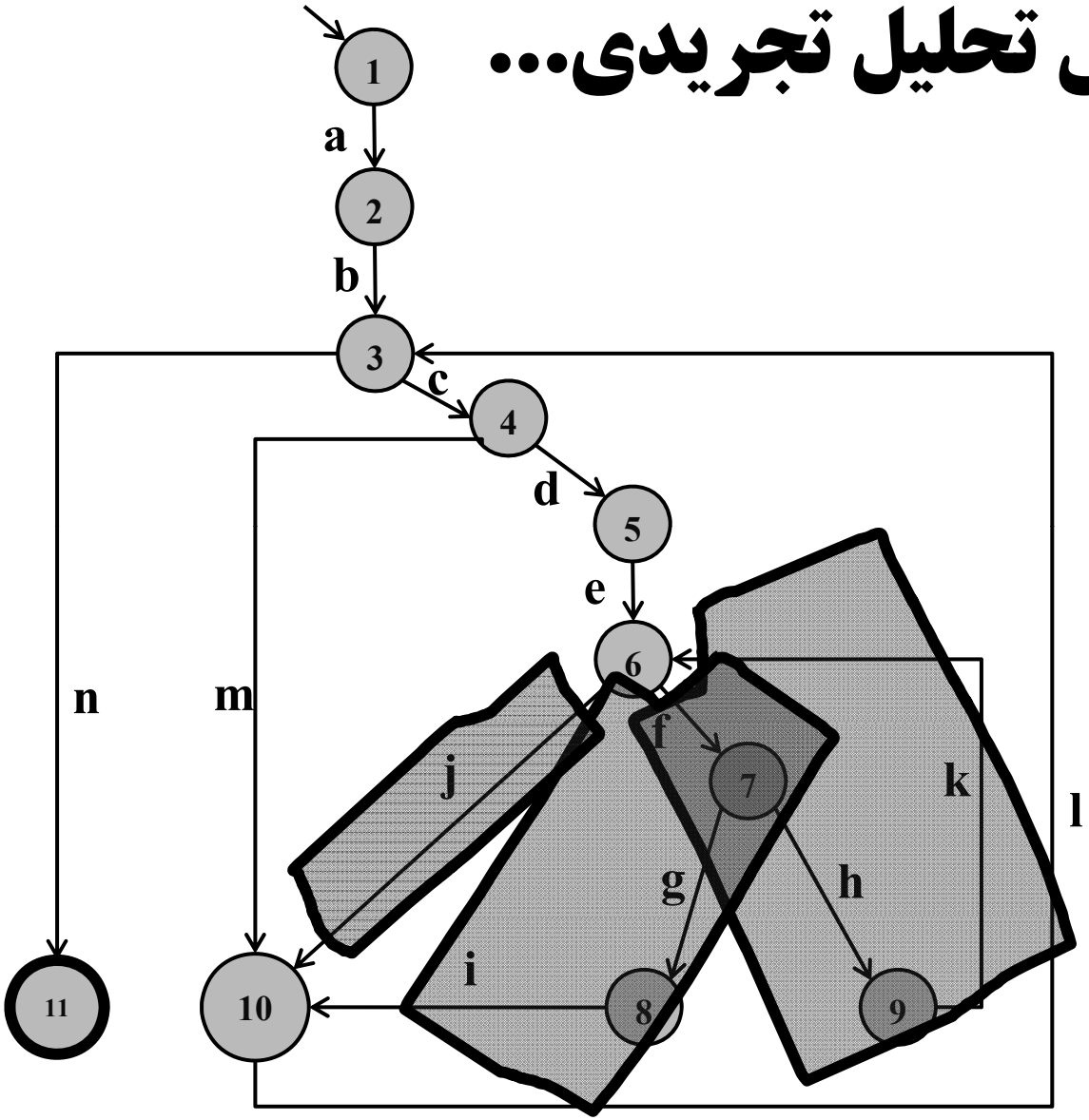
B=?



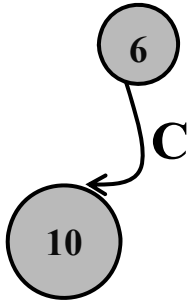
$\Rightarrow$B=deC

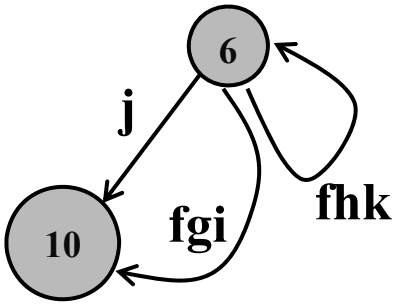
C=?

روش تحلیل تجریدی...



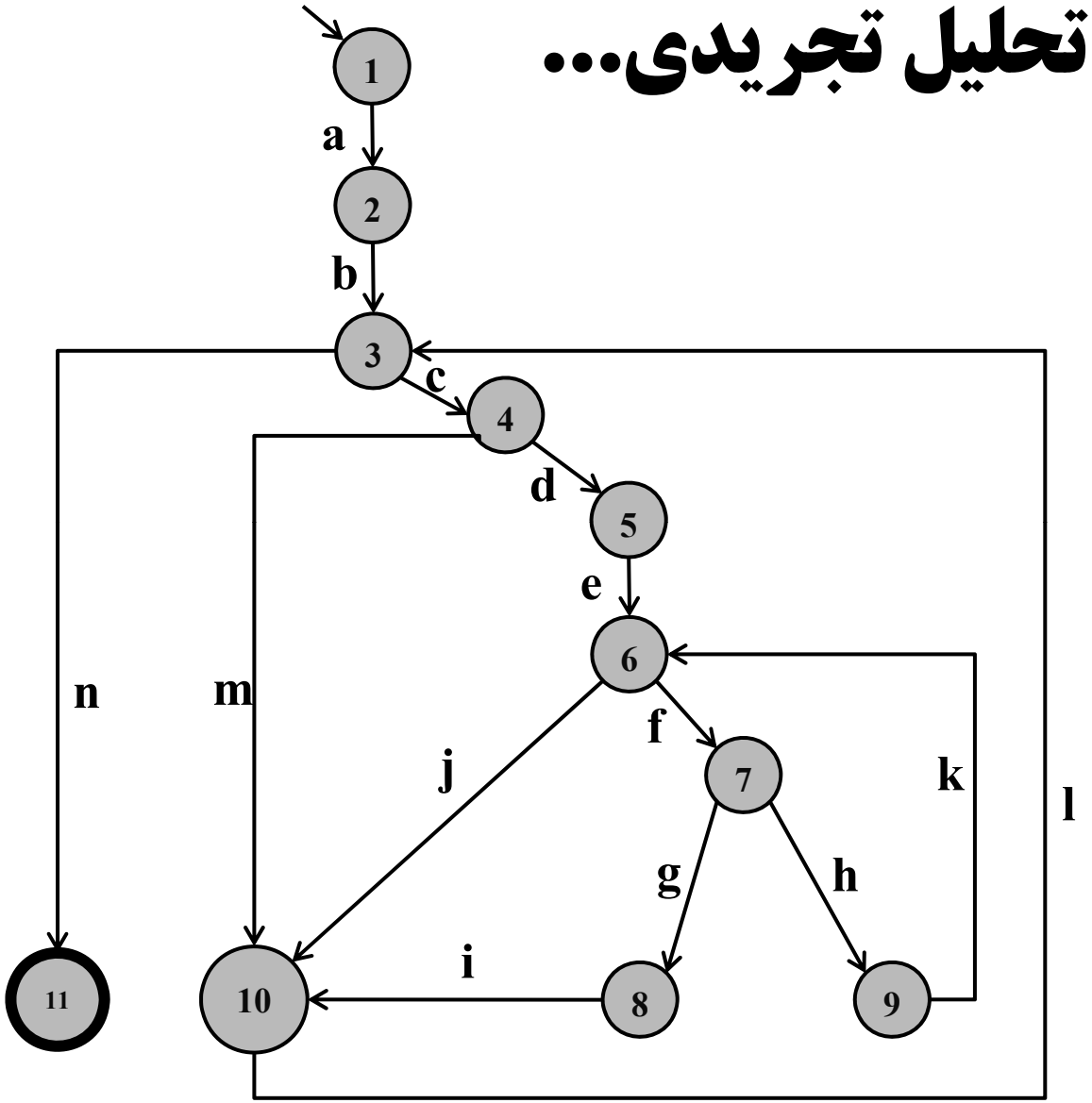
C=?





$\Rightarrow C=(fhk)^* (j+fgi)$

روش تحلیل تجریدی...



- پس با جایگذاری:

P.E. = abX*n

X=cAl

A=m+B

B=deC

C=(fhk)* (j+fgi)

⇒

P.E. = ab (cAl)* n

= ab (c (m+B) l)* n

= ab (c (m+deC) l)* n

= ab (c (m+de(fhk)*(j+fgi)) l)* n

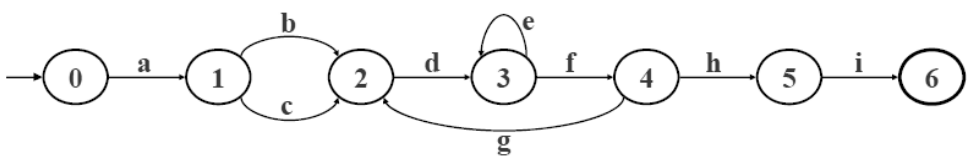
کاربردهای عبارت مسیر (PE)

- (۱) استخراج ورودی های آزمون
- (۲) تعیین تعداد مسیرهای یک گراف
- (۳) تعیین طول بلندترین مسیر گراف
- (۴) تعیین مینیمم تعداد مسیرها برای پوشش یال
- (۵) تحلیل عمل های مکمل

کاربرد(۱): استخراج ورودی های آزمون

- بسیار ساده است: باید گراف پوشش داده شود.
- پس کافی است هر حاصلضرب مسیر را پوشش دهیم
- حلقه ها با «توان» های موجود در عبارت مسیر مشخص می شوند. اگر * وجود داشته باشد یعنی تعداد گردش در حلقه فاقد محدوده است؛ باید آن را (بر اساس شناختی که از برنامه داریم) با جایگزینی اعداد ثابت محدود کنیم

- برای مثال اول:



- عبارت مسیر نهایی: abde*f (gde*f)* hi + acde*f (gde*f)* hi
- نیازمندیهای آزمون: TR= { $abde^5f\ (gde^5f)^5\ hi$, $acde^5f\ (gde^5f)^5\ hi$ }

کاربرد(۲): تعیین تعداد مسیرهای گراف

- وجود حلقه ها یعنی تعداد مسیرها بینهایت است.
- پس باید حلقه ها را به میزانی معقول محدود کنیم (به تشخیص آزمونگر).
- به جای * عدد ثابتی بگذاریم که نشان دهنده ماکسیمم تعداد تکرار حلقه باشد.

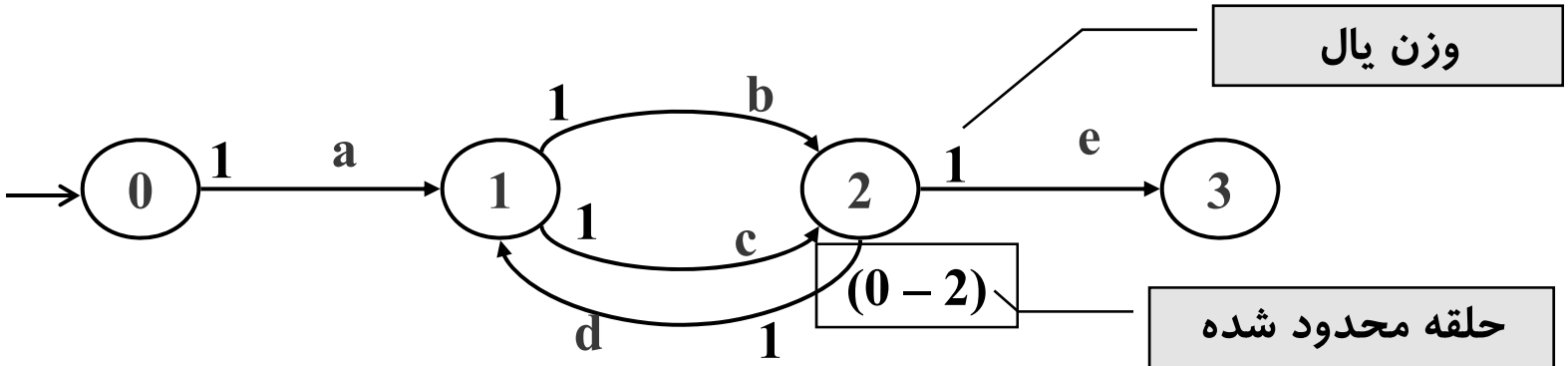
- در این روش به مسیرها وزنی داده می شود (به نوعی نشان دهنده هزینه اجرای مسیر).
- وزن یال به صورت پیش فرض ۱ است (می تواند به تشخیص آزمونگر عدد دیگری باشد)
- وزن حلقه: ماکسیمم تعداد تکرار مجاز

- با محاسبه وزن مسیر، تعداد تخمینی مسیرها به دست می آید (تخمینی به خاطر وجود حلقه ها است):

عنصر در عبارت مسیر	نحوه محاسبه
عبارت مسیر به صورت A+B	وزن A + وزن B
عبارت مسیر به صورت AB	وزن A × وزن B
حلقه دارای m تا n بار تکرار (و وزن w)	جمع وزن همه تکرار ها = $\sum_{i=m}^{n} w^i$

کاربرد(۲): تعیین تعداد مسیرهای گراف...

- مثال:



حلقه محدود شده است: مینیمم ۰ و ماکسیمم ۲ بار تکرار

$$pe = a\,(b + c)\,(d\,(b + c))^{\underline{2}}\,e$$

$$= 1 * (1 + 1) * (1 * (1 + 1))^{\underline{2}} * 1$$

$$= 1 * 2 * 2^{\underline{2}} * 1$$

$$= 2 * (\Sigma^{2}_{i=0}\, 2^{i}) * 1$$

$$= 2 * (2^{0} + 2^{1} + 2^{2}) * 1$$

$$= 2 * (1 + 2 + 4) * 1$$

$$= 2 * 7 * 1$$

$$= 14$$

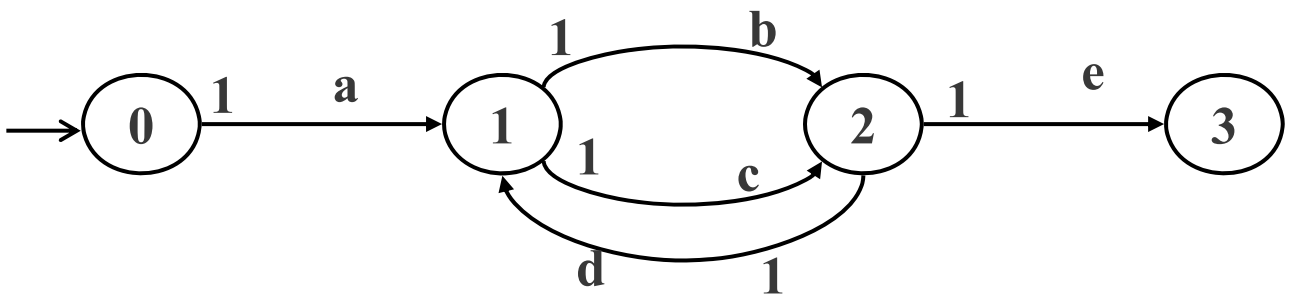
عنصر در عبارت مسیر	نحوه محاسبه
عبارت مسیر به صورت A+B	وزن A + وزن B
عبارت مسیر به صورت AB	وزن A ✕ وزن B
حلقه	جمع وزن همه تکرار ها

کاربرد(۳): تعیین طول بلندترین مسیر گراف

- به صورت زیر محاسبه می شود:

عنصر در عبارت مسیر	نحوه محاسبه
عبارت مسیر به صورت A+B	ماکسیمم (وزن A ، وزن B)
عبارت مسیر به صورت AB	وزن A + وزن B
حلقه به صورت $A^{\underline{n}}$	وزن A × n

- مثال:



$$pe = a\,(b + c)\,(d\,(b + c))^{\underline{2}}\,e$$

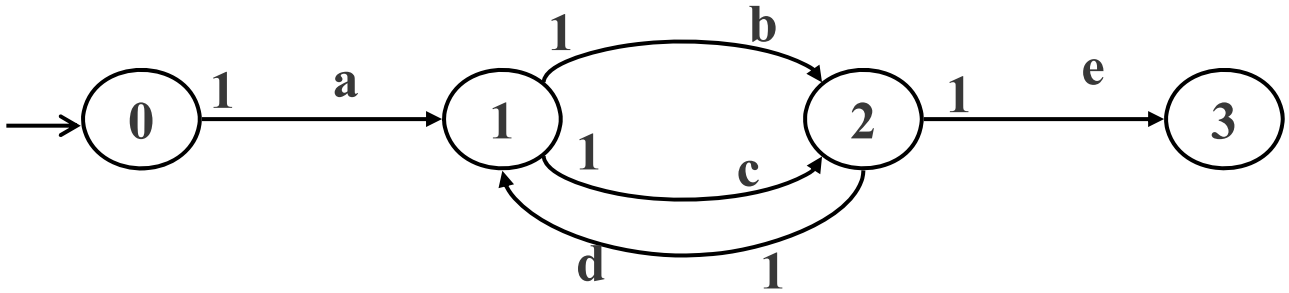
$$1 + \max(1,1) + 2 * (1 + \max(1,1)) + 1 = 7$$

کاربرد(۴): تعیین مینیمم تعداد مسیرها برای پوشش یال

- **به صورت زیر محاسبه می شود:**

عنصر در عبارت مسیر	نحوه محاسبه
عبارت مسیر به صورت A+B	وزن A + وزن B
عبارت مسیر به صورت AB	ماکسیمم (وزن A ، وزن B)
حلقه به صورت $A^{\underline{n}}$	۱ (اگر همه مسیرها را می توان با یک آزمون، آزمایش کرد) یا وزن A (در غیر این صورت)

- **مثال:**



$$pe = a\,(b + c)\,(d\,(b + c))^{\underline{2}}\,e$$

$$minPathNo = \max(1, 2, 1, 2, 1) = 2$$

خلاصه کاربردهای (۲)، (۳) و (۴)

کاربرد (۲):
تعیین حداکثر تعداد مسیرهای گراف

عنصر در عبارت مسیر	نحوه محاسبه
عبارت مسیر به صورت A+B	وزن A + وزن B
عبارت مسیر به صورت AB	وزن A × وزن B
حلقه دارای m تا n بار تکرار (و وزن w)	جمع وزن همه تکرار ها = $\sum_{i=m}^{n} w^i$

کاربرد (۳):
تعیین طول بلندترین مسیر گراف

عنصر در عبارت مسیر	نحوه محاسبه
عبارت مسیر به صورت A+B	ماکسیمم (وزن A ، وزن B)
عبارت مسیر به صورت AB	وزن A + وزن B
حلقه به صورت $A^{\underline{n}}$	وزن A × n

کاربرد (۴):
تعیین حداقل تعداد مسیرهای گراف برای پوشش یال

عنصر در عبارت مسیر	نحوه محاسبه
عبارت مسیر به صورت A+B	وزن A + وزن B
عبارت مسیر به صورت AB	ماکسیمم (وزن A ، وزن B)
حلقه به صورت $A^{\underline{n}}$	۱ (اگر همه مسیرها را می توان با یک آزمون، آزمایش کرد) یا وزن A (در غیر این صورت)

کاربرد(۵): تحلیل عمل های مکمل

- روشی برای یافتن ناهنجاری های احتمالی است؛ مثلاً:
  - ناهنجاری def-use (این که use قبل از def خود واقع شود)
  - ناهنجاری ADT فایل (بستن فایل قبل از باز کردن آن) و ...

- دو عمل، مکمل هستند اگر رفتار آنها نقیض هم باشد؛ یا یکی باید قبل از دیگری انجام شود.
  - push و pop
  - enqueue و dequeue
  - تخصیص حافظه و آزاد کردن حافظه
  - open و close

- به جای وزن یال ها، یکی از سه برچسب زیر را قرار می دهیم.
  - C یا عمل ایجاد کننده
  - D یا عمل تخریب کننده
  - ۱ یعنی هیچکدام

کاربرد(۵): تحلیل عمل های مکمل...

- برای ضرب و جمع از جدول های زیر استفاده می کنیم:

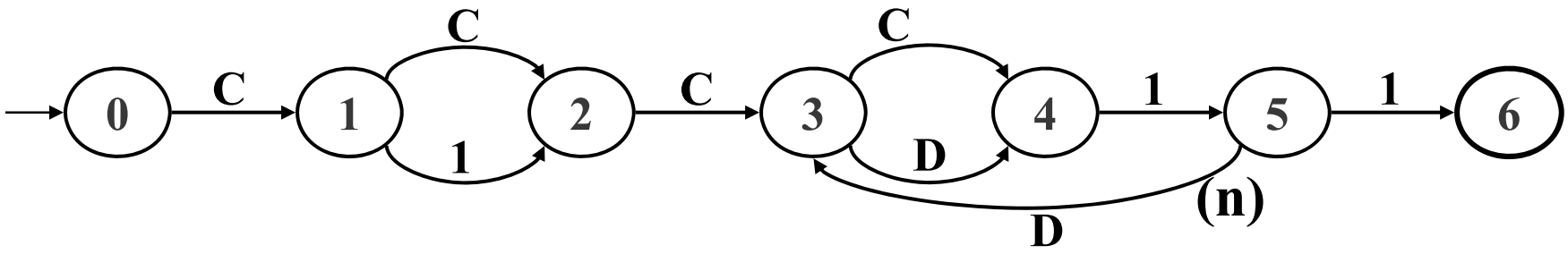
*	C	D	1
C	C^2	1	C
D	DC	D^2	D
1	C	D	1

+	C	D	1
C	C	C+D	C+1
D	D+C	D	D+1
1	C+1	D+1	1

- با جبر معمولی کمی فرق دارد:
  - $C*D = 1$, $C+C = C$, $D+D = D$
  - ضرب خاصیت جابجایی ندارد زیرا
  - تخریب کننده می تواند اثر ایجاد کننده را خنثی کند ($CD=1$)
  - ولی نه برعکس ($DC\neq1$)

کاربرد(۵): تحلیل عمل های مکمل...

- مثال:



$$pe = C\ (C+1)\ C\ (C+D)\ 1\ (D\ (C+D)\ 1)^n\ 1$$

$$= (CCCC + CC\cancel{CD} + CCC + C\cancel{CD})\ (DC + DD)^n$$

$$C*D = 1$$

$$\Rightarrow pe = (CCCC + CC + CCC + C)\ (DC + DD)^n$$

کاربرد(۵): تحلیل عمل های مکمل...

$$pe = (CCCC + CC + CCC + C)\ (DC + DD)^n$$

- دو سوال می پرسیم:
- (۱) آیا بر اساس عبارت مسیرمان می توانیم تخریب کننده های بیشتری از ایجاد کننده ها داشته باشیم؟
- مثلاً در شرایط زیر این اتفاق می افتد:

  - $CCCD\ (DD)^n,\ n > 1$
  - $CCCD\ (DD)^n,\ n > 0$
  - $CCC\ (DDDCDD)$

- (۲) آیا بر اساس عبارت مسیرمان می توانیم ایجاد کننده های بیشتری از تخریب کننده ها داشته باشیم؟
- مثلاً:

  - $CCCC$
  - $CCD\ (DC)^n$

- هر پاسخ مثبت به این پرسشها نشان دهنده مشخصات آزمون برای کشف یک ناهنجاری بالقوه است.

پایان جلسه پنجم