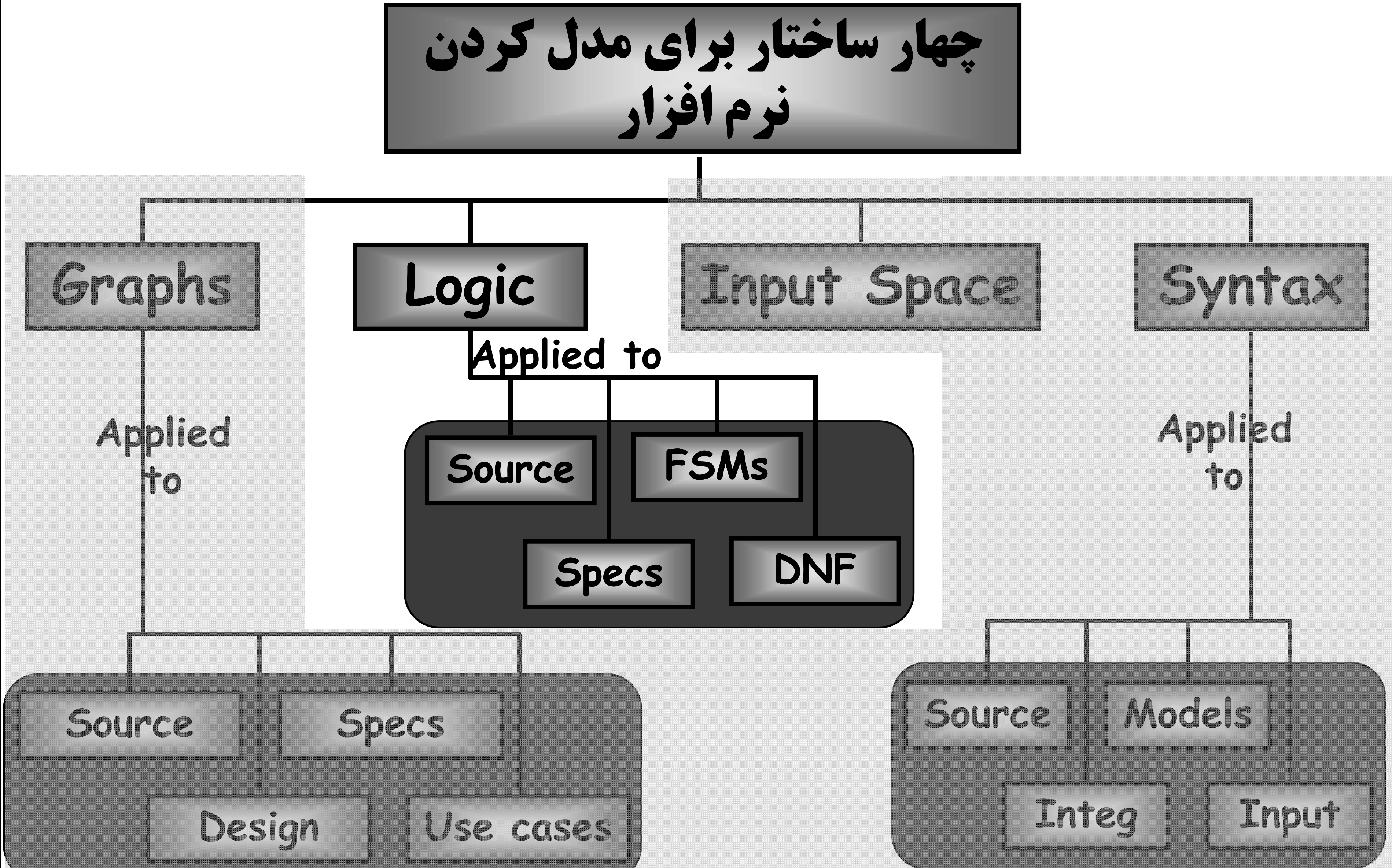
آزمون نرم افزار - بخش ۳-۱ و ۳-۲

پوشش منطق

صدیقه خوشنویس
دانشگاه آزاد اسلامی - واحد شهرقدس

پوشش منطق


چهار ساختار برای مدل کردن
نرم افزار
Graphs
Logic
Input Space
Syntax
Applied to
Source
FSMs
Specs
DNF
Applied to
Source
Specs
Design
Use cases
Applied to
Source
Models
Integ
Input

پوشش عبارتهای منطقی

- در بسیاری از موقعیتها (مثل شرط های برنامه) با عبارتهای منطقی رو به رو می شویم.

- پوشش عبارات منطقی در بعضی از سازمانها در آمریکا مانند هوانوردی، در آزمایش نرم‌افزارهای حساس به ایمنی، الزامی است.

- در پوشش منطق، هدف آن است که زیرمجموعه ای از جدول درستی این عبارتها را انتخاب و آنها را آزمایش کنیم.

گزاره (predicate) و جزء (clause)

- **گزاره (predicate) : عبارتی که دارای ارزش بولین (یعنی True یا False) باشد.**

- **درون یک گزاره می توانیم داشته باشیم:**
  - **متغیرهای بولین - فراخوانی توابع بولین**
  - **متغیرهای غیربولین که با هم دارای رابطه $>, <, ==, >=, <=, !=$ باشند**

- **بین متغیرهای بولین در یک گزاره می توان از اپراتورهای زیر استفاده کرد:**
  - **$\neg$: اپراتور نقیض کردن**
  - **$\wedge$: اپراتور and**
  - **$\vee$: اپراتور or**
  - **$\rightarrow$: اپراتور استلزام (اگر-آنگاه)**
  - **$\oplus$: اپراتور یای انحصاری (xor)**
  - **$\leftrightarrow$: اپراتور هم ارزی**

- **جزء (clause) : گزاره ای بدون اپراتور منطقی است.**

یادآوری اپراتورهای منطقی

a	¬a
T	F
F	T

$\neg T=F$
$\neg F=T$
$\neg(\neg a)=a$

ارزش متغیر را معکوس (نقیض) می کند

a	b	a→b
T	T	T
T	F	F
F	T	T
F	F	T

$a \rightarrow b = \neg a \vee b$
$a \rightarrow T = T$
$a \rightarrow F = \neg a$
$a \rightarrow a = T$
$a \rightarrow \neg a = \neg a$
$T \rightarrow a = a$
$F \rightarrow a = T$

همواره T است بجز وقتی که از فرض T به نتیجه F رسیده باشیم

قوانین دمرگان
$\neg(a\wedge b) = \neg a \vee \neg b$
$\neg(a\vee b) = \neg a \wedge \neg b$

a	b	a∧b
T	T	T
T	F	F
F	T	F
F	F	F

$a \wedge T=a$
$a \wedge F=F$
$a \wedge a=a$
$a \wedge \neg a=F$

وقتی T است که ارزش هر دو طرف T باشد

a	b	a↔b
T	T	T
T	F	F
F	T	F
F	F	T

$a \leftrightarrow T = a$
$a \leftrightarrow F = \neg a$
$a \leftrightarrow a = T$
$a \leftrightarrow \neg a = F$

$a \leftrightarrow b = \neg(a \oplus b)$
$a \leftrightarrow b = (a \wedge b) \vee (\neg a \wedge \neg b)$

وقتی T است که ارزش طرفین یکسان باشند

a	b	a⊕b
T	T	F
T	F	T
F	T	T
F	F	F

$a \oplus T = \neg a$
$a \oplus F = a$
$a \oplus a = F$
$a \oplus \neg a = T$

$a \oplus b = \neg(a \leftrightarrow b)$
$a \oplus b = (a \wedge \neg b) \vee (\neg a \wedge b)$

وقتی T است که ارزش طرفین متفاوت باشند

a	b	a∨b
T	T	T
T	F	T
F	T	T
F	F	F

$a \vee T=T$
$a \vee F=a$
$a \vee a=a$
$a \vee \neg a=T$

وقتی T است که ارزش حداقل یکی از طرفین T باشد

اولویت: به ترتیب: پرانتز، ¬، ∧، ∨، →، و نهایتاً ↔ و ⊕ دارای اولویت یکسان هستند.

چند مثال

$(a < b) \vee f(z) \wedge D \wedge (m >= n*o)$

- گزاره بالا چهار جزء (کلاز) دارد:
  - (a < b) : عبارت رابطه ای (دو متغیر غیربولین که با هم رابطه < دارند)
  - f(z) : فراخوانی تابع بولین
  - D : متغیر بولین
  - (m >= n*o) : عبارت رابطه ای

- اغلب گزاره ها دارای اجزاء (کلازهای) کمی هستند.

- منبع گزاره ها:
  - نقاط تصمیم در کد برنامه ها (شرط ها) یا در نمودار فعالیت UML (جعبه های تصمیم)
  - شرط ها (guard) در FSM ها و نمودارهای State Chart
  - نیازمندی های نرم افزار
  - شرط های WHERE در queryهای SQL

کاربرد گزاره در آزمایش

- روش کلی در آزمایش گزاره ها:
  - ایجاد مدلی از نرم افزار به صورت یک یا چند گزاره
  - ایجاد مجموعه نیازمندیهای آزمون (TR) برای برآورده کردن ترکیبی از اجزاء (کلازها)

- مخفف هایی که به کار خواهیم برد:
  - P : مجموعه گزاره ها
  - p : یک گزاره منفرد در P
  - C : مجموعه اجزاء کل گزاره های P
  - C_p : مجموعه اجزاء گزاره منفرد p
  - c : یک جزء منفرد در C

پوشش گزاره و جزء

- اولین و ساده ترین معیارهای آزمایش منطق، نیازمند آن هستند که هر گزاره و هر جزء یک بار True و یک بار False شوند.

<u>پوشش گزاره (PC)</u> : برای هر p در TR ،P دارای دو نیازمندی است: ارزش p، True باشد، و ارزش p، False باشد.

- پوشش گزاره (PC)، همه جزء ها را ارزیابی نمی کند، پس:

<u>پوشش جزء (CC)</u> : برای هر c در TR ،C دارای دو نیازمندی است: ارزش c، True باشد، و ارزش c، False باشد.

مثال از پوشش گزاره

برای گزاره : $((a < b) \vee D) \wedge (m >= n*o)$

گزاره= true

$a = 5, b = 10, D = true, m = 1, n = 1, o = 1$

$= (5 < 10) \vee true \wedge (1 >= 1*1)$

$= true \vee true \wedge TRUE$

$= true$

گزاره= false

$a = 10, b = 5, D = false, m = 1, n = 1, o = 1$

$= (10 < 5) \vee false \wedge (1 >= 1*1)$

$= false \vee false \wedge TRUE$

$= false$

با توجه به TRUE ماندن ارزش کلاز آخر، مشاهده می کنیم که ارزش همه کلازها در پوشش گزاره لزوماً تغییر نمی کند. پس هر دو حالت T و F برای همه کلازها با پوشش گزاره تست نمی شود. پس به پوشش کلاز (جزء) نیاز داریم.

مثال از پوشش جزء

برای گزاره : $((a < b) \vee D) \wedge (m >= n*o)$

(a < b) = true	(a < b) = false	D = true	D = false
a = 5, b = 10	a = 10, b = 5	D = true	D = false

m >= n*o = true	m >= n*o = false
m = 1, n = 1, o = 1	m = 1, n = 2, o = 2

حالتهای true

حالتهای false

دو آزمون

1) a = 5, b = 10, D = true, m = 1, n = 1, o = 1

2) a = 10, b = 5, D = false, m = 1, n = 2, o = 2

مشکل پوشش گزاره و پوشش جزء

- PC لزوماً CC را در خود ندارد (زیرا هر دو حالت T و F را برای همه اجزاء پوشش نمی دهد).
  - پس عملاً اجزاء منفرد تست نمی شوند!

- CC هم لزوماً PC را در خود ندارد.
  - یعنی ممکن است با تغییر ارزش جزء، ارزش گزاره تغییر نکند.
  - و ما قطعاً این را نمی خواهیم!
  - زیرا در این صورت تغییر ارزش جزء هیچ اثر بیرونی نخواهد داشت و در زمان اجرا اثر آن گم می شود و کاملاً بلا استفاده خواهد بود!
  - پس گزاره های اصلی با پوشش جزء لزوماً تست نمی شوند.

- ساده ترین راه حل آن است که همه ترکیبهای ممکن اجزاء را تست کنیم تا هم همه اجزاء و هم گزاره اصلی تست شود...

پوشش ترکیباتی (combinatorial) یا CoC

- نام دیگر آن پوشش «چند شرطی» (Multiple Condition) است.
- CoC نیازمند همه ترکیبهای ممکن از مقادیر کلازهای گزاره است(یعنی جدول درستی کامل گزاره)

<u>پوشش ترکیباتی (CoC)</u>: برای هر p در TR ،P شامل این نیازمندی است که همه کلازهای C_p دارای همه ترکیبهای ممکن ارزشها در جدول درستی باشند.

	a < b	D	m >= n*o	$((a < b) \lor D) \land (m >= n*o)$
1	T	T	T	T
2	T	T	F	F
3	T	F	T	T
4	T	F	F	F
5	F	T	T	T
6	F	T	F	F
7	F	F	T	F
8	F	F	F	F

مشکل پوشش ترکیباتی

- این پوشش، ساده، جامع و آسان است.
- ولی بسیار گران است.
- اگر تعداد کلازها N باشد، آنگاه نیاز به 2^N آزمون داریم.
  - اگر تعداد کلازها بیشتر از ۳ یا ۴ شود، غیرکاربردی خواهد بود.
- راه حل اصلی:

کاری کنیم که تغییر ارزش هر کلاز موجب تغییر ارزش گزاره شود.

- در این صورت می گوییم کلاز را «فعال کرده ایم».
- کلاز فعال، تعیین کننده (دترمینان) گزاره است.
- چنین کلازی، «مستقل» از سایر کلازها روی گزاره اثر خواهد گذاشت.

فعال کردن کلاز و مفهوم دترمینان

تعیین کردن (دترمینان بودن): برای کلاز c_i در گزاره p، می گوییم c_i، p را تعیین می کند (یا دترمینان p است) اگر و فقط اگر مقادیر سایر کلازهای c_j طوری باشد که تغییر ارزش c_i موجب تغییر ارزش p شود.

- فعال کردن c_i مورد نظر نیازمند تنظیم ارزش سایر کلازها (c_j ها) است.

- به کلاز مورد نظر (مثلاً جهت فعال یا غیرفعال کردن) (c_i)، کلاز اصلی (major)،
- و به سایر کلازها (c_jها)، کلاز فرعی (minor) می‌گوییم.

- هدف: برای حالتی که کلاز، دترمینان گزاره است، آزمونهایی را بیابیم.
  - چند معیار وجود دارد.
- ابتدا دو مثال از تعیین کنندگی (دترمینان بودن) کلاز:

دو مثال از کلاز فعال یا دترمینان

مثال ۱:

$$P = A \wedge B$$

می خواهیم A دترمینان P باشد؛ باید ارزش B را برای این هدف تنظیم کنیم:

اگر مقدار ***B=false*** باشد، مقدار A هرچه باشد، بی اثر است و حاصل همیشه F است. پس در این حالت A دترمینان P نیست.

ولی اگر ارزش ***B=true*** باشد، A دترمینان P خواهد بود زیرا با تغییر ارزش آن، ارزش P تغییر می کند.

مثال۲:

$$P = A \vee B$$

می خواهیم A دترمینان P باشد؛ باید ارزش B را برای این هدف تنظیم کنیم:

اگر مقدار ***B=true*** باشد، مقدار A هرچه باشد، بی اثر است و حاصل همیشه T است. پس در این حالت A دترمینان P نیست.

ولی اگر ارزش ***B=false*** باشد، A دترمینان P خواهد بود زیرا با تغییر ارزش آن، ارزش P تغییر می کند.

- در مثال بالا وقتی A را برای فعال (دترمینان) کردن در نظر می گیریم، A کلاز اصلی، و B کلاز فرعی خواهد بود.

- سوال: اگر در دو مثال بالا بخواهیم B دترمینان باشد، باید ارزش A را چطور تنظیم کنیم؟

پوشش جزء فعال (ACC)

<u>پوشش جزء فعال (ACC)</u>: برای هر گزاره (p) و برای هر کلاز اصلی c_i در C_p، ارزش کلازهای فرعی c_j را (که j!=i) طوری انتخاب کنید که c_i دترمینان p باشد. در این صورت TR <u>برای هر</u> c_i دارای دو نیازمندی است: c_i برابر با True باشد، و c_i برابر با False باشد.

مثال: برای گزاره

$p= a \vee b$

داریم:

وقتی ci=a دترمینان و کلاز اصلی، و cj=b کلاز فرعی است

$p = a \vee b$	
a	b
T	F
F	F

وقتی ci=b دترمینان و کلاز اصلی، و cj=a کلاز فرعی است

$p = a \vee b$	
a	b
F	T
F	F

تکراری

=> TR={ (a=T,b=F) , (a=F,b=F) , (a=F,b=T) , ~~(a=F,b=F)~~ }

- <u>ابهام</u>: وقتی کلاز اصلی T و F می شود، آیا کلازهای فرعی باید مقادیرشان ثابت بماند؟ یا اجازه دارد تغییر کند؟ (در مثال بالا به ناچار ثابت مانده است ولی گاهی می تواند متغیر هم باشد)

رفع ابهام در ACC

<u>$p = a \vee (b \wedge c)$</u>

Major clause : a

a = true, b = false, c = true

a = false, b = false, c = false

ابهام: آیا تغییر c مجاز است؟

- **سه راه حل برای رفع این ابهام پیشنهاد شده است:**
  1. **کلازهای فرعی نیازی نیست ثابت بمانند (General ACC = GACC).**
  2. **کلازهای فرعی باید ثابت بمانند (Restricted ACC = RACC).**
  3. **کلازهای فرعی باید منجر به تغییر ارزش گزاره شوند (Correlated ACC=CACC).**

ACC کلّی یا GACC

<u>پوشش جزء فعال کلی (GACC) :</u> همان ACC است که در آن با تغییر ارزش جزء اصلی، ارزش جزء های فرعی نیازی نیست ثابت بماند (می تواند ثابت بماند یا ثابت نماند).

یعنی برای همه cjها: $cj(ci=T) = cj(ci=F)$ یا $cj(ci=T) \neq cj(ci=F)$

- احتمال آن وجود دارد که با GACC، پوشش گزاره محقق نشود!
- در حالی که هدف این بود که با تغییر ارزش کلازها، ارزش گزاره هم تغییر کند تا هم کلازها و هم گزاره تست شوند.

ACC محدود شده یا RACC

<u>پوشش جزء فعال محدود شده (RACC) :</u> همان ACC است که در آن با تغییر ارزش جزء اصلی، ارزش جزء های فرعی باید ثابت بماند.

یعنی برای همه cjها: cj(ci=T) = cj(ci=F)

- اغلب منجر به نیازمندی‌های آزمونی می شود که برخی از آنها ناممکن هستند.
- هیچ دلیل منطقی برای چنین محدود کردنی وجود ندارد (می خواهیم کلاز دترمینان روی ارزش گزاره اصلی تاثیر بگذارد؛ ولی چرا با این نوع محدودیت؟).

همبسته یا ACC CACC

پوشش جزء فعال همبسته (CACC) : همان ACC است که در آن با تغییر ارزش جزء اصلی، ارزش جزء های فرعی باید طوری باشد که الزاماً ارزش گزاره را تغییر دهد.

یعنی برای همه cjها: $p(ci=T) \neq p(ci=F)$

- جدیدتر و منطقی تر است.
- چون قید نشده، نیازی نیست که مقدار کلازهای فرعی ثابت بماند (مثل GACC)
- نسبت به GACC یک قید مهم دارد: این که حتماً ارزش گزاره تغییر کند.
- واضح است که پوشش گزاره را در بر دارد (آن را subsume می کند).

یک مثال عمومی

اگر a کلاز اصلی باشد:

	a	b	c	$a \wedge (b \vee c)$
1	T	T	T	T
2	T	T	F	T
3	T	F	T	T
5	F	T	T	F
6	F	T	F	F
7	F	F	T	F

برای GACC: TR = انتخاب زوج سطرهایی که در آنها سطر اول یکی از سطرهای ۱یا ۲یا ۳ و دومی یکی از سطرهای ۵ یا ۶ یا ۷ باشد (جمعاً ۹ زوج سطر)

برای CACC هم همینطور چون با انتخاب این زوج سطرها، مقدار p تغییر می کند.

اگر a کلاز اصلی باشد:

	a	b	c	$a \wedge (b \vee c)$
1	T	T	T	T
5	F	T	T	F
2	T	T	F	T
6	F	T	F	F
3	T	F	T	T
7	F	F	T	F

برای RACC، TR فقط شامل سه زوج سطر بالا است (دقت کنید که ارزش b و c در هر زوج سطر ثابت مانده است)

توجه: در سطرهای ۴ و ۸ که در آنها b و c هردو F هستند، a دترمینان نیست؛ به همین دلیل این دو سطر از دو جدول درستی فوق حذف شده اند.

روش استخراج TR برای پوششهای ACC

- ۱. برای هر گزاره p، لیست کلازهای آن را مشخص می کنیم.
- ۲. برای هر کلاز c مشخص می کنیم که در چه شرایطی، دترمینان p است (تعیین p_c).
- ۳. جدول درستی را رسم می کنیم که شامل کلازها، خود p، و به ازای هر c، p_c است.
- ۴. هر بار یکی از کلازها را دترمینان می گیریم و با توجه به مطالب قبلی، نیازمندیهای آزمون GACC، RACC یا CACC را استخراج می کنیم.

روش دترمینان کردن یک کلاز

- تنظیم کلازهای فرعی برای دترمینان شدن یک کلاز اصلی، برای گزاره های کوچک آسان است.
- ولی برای گزاره های کمی بزرگتر دشوار است.
- روش کلی به این صورت است:

- p_c = شرایطی که در آن c دترمینان p است.
- $p_{c=true}$= عبارتی که با جایگذاری T به جای c در گزاره p به دست می آید.
- $p_{c=false}$ = عبارتی که با جایگذاری F به جای c در گزاره p به دست می آید.

$$p_c = p_{c=true} \oplus p_{c=false}$$

- عبارتی که به دست می آید مشخص می کند که کلازهای فرعی کدام ها باید T و کدام ها باید F باشند.
  - هر کدام که not نداشت: True
  - هر کدام که not داشت: False

یادآوری درباره XOR (⊕)

- جدول درستی:

a	b	p = a ⊕ b
T	T	F
T	F	T
F	T	T
F	F	F

- حاصل ⊕ وقتی T است که طرفین با هم متفاوت باشند.
- حاصل ⊕ وقتی F است که طرفین با هم یکسان باشند.

- چند رابطه بسیار مهم:

- $a \oplus a = F$
- $a \oplus \neg a = T$
- $a \oplus T = \neg a$
- $a \oplus F = a$
- $a \oplus b = (a \wedge \neg b) \vee (\neg a \wedge b)$

مثال از دترمینان کردن یک کلاز (۱)

$p = a \wedge b$

$$
\begin{aligned}
p_a &= p_{a=true} \oplus p_{a=false} \\
&= (true \wedge b) \oplus (false \wedge b) \\
&= b \oplus false \\
&= b
\end{aligned}
$$

یعنی b باید T باشد تا a دترمینان p شود.

$p = a \vee b$

$$
\begin{aligned}
p_a &= p_{a=true} \oplus p_{a=false} \\
&= (true \vee b) \text{ XOR } (false \vee b) \\
&= true \text{ XOR } b \\
&= \neg b
\end{aligned}
$$

یعنی b باید F باشد تا a دترمینان p شود.

$p = a \vee (b \wedge c)$

$$
\begin{aligned}
p_a &= p_{a=true} \oplus p_{a=false} \\
&= (true \vee (b \wedge c)) \oplus (false \vee (b \wedge c)) \\
&= true \oplus (b \wedge c) \\
&= \neg (b \wedge c) \\
&= \neg b \vee \neg c
\end{aligned}
$$

یعنی باید یا b یا c و یا هر دو آنها F باشند تا a دترمینان p شود.

مثال از دترمینان کردن یک کلاز (۲)

$$\underline{p = (a \wedge b) \vee (a \wedge \neg b)}$$

$$\begin{aligned} p_a &= p_{a=true} \oplus p_{a=false} \\ &= ((true \wedge b) \vee (true \wedge \neg b)) \oplus ((false \wedge b) \vee (false \wedge \neg b)) \\ &= (b \vee \neg b) \oplus false \\ &= true \oplus false \\ &= true \end{aligned}$$

یعنی a همواره دترمینان p است.

$$\underline{p = (a \wedge b) \vee (a \wedge \neg b)}$$

$$\begin{aligned} p_b &= p_{b=true} \oplus p_{b=false} \\ &= ((a \wedge true) \vee (a \wedge \neg true)) \oplus ((a \wedge false) \vee (a \wedge \neg false)) \\ &= (a \vee false) \oplus (false \vee a) \\ &= a \oplus a \\ &= false \end{aligned}$$

یعنی b هرگز نمی تواند دترمینان p باشد.

پوشش جزء غیرفعال (ICC)

- در پوشش جزء غیرفعال می خواهیم کاری کنیم که کلاز اصلی روی گزاره تاثیری نداشته باشد!

پوشش جزء غیرفعال (ICC): برای هر گزاره (p) و برای هر کلاز اصلی c_i در C_p، ارزش کلازهای فرعی c_j را (که j!=i) طوری انتخاب کنید که c_i دترمینان p نباشد. در این صورت TR برای هر c_i دارای دو نیازمندی است: c_i برابر با True باشد، و c_i برابر با False باشد.

برای هر ci، TR دارای چهار نیازمندی است:

(۱) ci=T باشد و p=T

(۲) ci=F باشد و p=T

(۳) ci=T باشد و p=F

(۴) ci=F باشد و p=F

پوشش جزء غیرفعال کلّی (GICC) و محدود شده (RICC)

- بر خلاف ACC مفهوم همبسته بودن دیگر معنی دار نیست (CICC نداریم) زیرا کلاز مورد نظر دیگر دترمینان نیست که بتواند ارزش گزاره را تغییر دهد.
- البته با توجه به چهار حالتی که در اسلاید قبل برشمردیم، پوشش گزاره همواره تضمین می‌شود (نیازی به حالتی شبیه همبسته هم نداریم).

پوشش جزء غیرفعال کلی (GICC): همان ICC است که در آن ارزش کلازهای فرعی نیازی نیست ثابت بمانند. (می تواند ثابت بماند یا ثابت نماند).

یعنی برای همه cjها: cj(ci=T) = cj(ci=F) یا cj(ci=T) ≠ cj(ci=F)

پوشش جزء غیرفعال محدود شده (RICC): همان ICC است که در آن با تغییر ارزش جزء اصلی، ارزش جزء های فرعی باید ثابت بماند.

یعنی برای همه cjها: cj(ci=T) = cj(ci=F)

روش استخراج TR برای پوششهای ICC

- ۱. برای هر گزاره p، لیست کلازهای آن را مشخص می کنیم.
- ۲. برای هر کلاز c مشخص می کنیم که در چه شرایطی، دترمینان p نیست.
- ۳. جدول درستی را رسم می کنیم که شامل کلازها، خود p، و به ازای هر c، p_c است.
- ۴. هر بار یکی از کلازها را به عنوان کلاز اصلی غیرفعال در نظر می گیریم و با توجه به مطالب قبلی، نیازمندیهای آزمون GICC یا RICC را استخراج می کنیم.

یک مثال کامل – صورت مسأله

- گزاره ای داریم به شکل زیر

$$(x<y \vee done) \wedge list.contains(str)$$

- توضیح: تابع contains چک می کند که آیا رشته str در list وجود دارد یا خیر.

- TRپوشش‌های زیر را برای آن به دست آورید:
  - الف) گزاره(PC)
  - ب) جزء(CC)
  - پ) GACC
  - ت) RACC
  - ث) CACC
  - ج) GICC
  - چ) RICC
- ح) ضمناً برای پوشش گزاره (PC)، یک مجموعه آزمون هم برای آن به دست آورید.

یک مثال کامل – قدم اول، استخراج کلازها

- گزاره دارای سه کلاز است:

$$(\underline{x<y} \vee \underline{done}) \wedge \underline{list.contains(str)}$$

- مشخص است که **done** یک متغیر بولین، و **contains** یک تابع بولین است.

- کلازها را به ترتیب، **a**، **b** و **c** می نامیم پس داریم: $(a \vee b) \wedge c$

یک مثال کامل – TR پوشش گزاره (PC) و تستهای آن (الف و ح)

- گزاره $p= (a\vee b) \wedge c$
- الف) TR برای پوشش گزاره آن است که گزاره یک بار T و یک بار F شود؛ پس:

$$TR_{PC}=\{p=T , p=F\}$$

$$\Rightarrow TR_{PC}=\{(a=T, b=T, c=T) , (a=T, b=T, c=F)\}$$

- البته مقادیر دیگر هم برای کلازها امکان پذیر است تا *p* را T یا F کنیم.

- ح) دو آزمون نیاز داریم:

p	a	b	c
p=T	x=2 , y=3	done=True	list=["red", "blue", "green"] , str="red"
p=F	x=2 , y=3	done=True	list=["blue", "green"] , str="white"

یک مثال کامل – پوشش جزء (CC) (ب)

- گزاره p= $(a \vee b) \wedge c$
- الف) TR برای پوشش جزء یا کلاز آن است که هر کلاز یک بار T و یک بار F شود؛ پس:

$$TR_{CC}=\{a=T\ ,\ a=F\ ,\ b=T\ ,\ b=F\ ,\ c=T\ ,\ c=F\}$$

- طبق جدول زیر:
- علامت × به معنای don’t care است (فرقی ندارد T باشد یا F).
- چند نکته:
  - حداکثر به ۶ آزمون نیاز داریم.
  - شاید با یک آزمون چند نیازمندی را بتوان برآورده کرد:
  - مثلا اگر در آزمون ما هر سه کلاز برابر T باشند، سه سطر اول، سوم و پنجم برآورده می شود.
  - پس حداقل با دو آزمون می توان TR را برای پوشش CC برای این گزاره برآورده کرد.

tr	a	b	c
a=T	T	×	×
a=F	F	×	×
b=T	×	T	×
b=F	×	F	×
c=T	×	×	T
c=F	×	×	F

یک مثال کامل – پوشش GACC (پ)

- گزاره $p = (a \vee b) \wedge c$

- برای کلیه پوششهای جزء فعال و غیرفعال ابتدا باید به ازای هر کلاز t، p_t را به دست آوریم یعنی تعیین کنیم در چه شرایطی t دترمینان p هست (و در نتیجه مشخص می شود در چه شرایطی t دترمینان p نیست).
- فرمول انجام این کار: $p_t = p_{t=true} \oplus p_{t=false}$

- پس داریم:

- $p_a = p_{a=true} \oplus p_{a=false} = ((T \vee b) \wedge c) \oplus ((F \vee b) \wedge c) = (c) \oplus (b \wedge c)$
- $p_b = p_{b=true} \oplus p_{b=false} = ((a \vee T) \wedge c) \oplus ((a \vee F) \wedge c) = (c) \oplus (a \wedge c)$
- $p_c = p_{c=true} \oplus p_{c=false} = ((a \vee b) \wedge T) \oplus ((a \vee b) \wedge F) = (a \vee b) \oplus (F) = (a \vee b)$

یک مثال کامل – پوشش GACC (پ)...

- حالا یک جدول درستی به صورت زیر تشکیل می دهیم.

ردیف	a	b	c	$p = (a \vee b) \wedge c$	$p_a = c \oplus (b \wedge c)$	$p_b = c \oplus (a \wedge c)$	$p_c = (a \vee b)$
1	T	T	T	T			T
2	T	T	F	F			T
3	T	F	T	T	T		T
4	T	F	F	F			T
5	F	T	T	T		T	T
6	F	T	F	F			T
7	F	F	T	F	T	T	
8	F	F	F	F			

یک مثال کامل – پوشش GACC (پ)...

2 ← برای a → 1

ردیف	a	b	c	p	p_a	p_b	p_c
1	T	T	T	T			T
2	T	T	F	F			T
3	T	F	T	T	T		T
4	T	F	F	F			T
5	F	T	T	T		T	T
6	F	T	F	F			T
7	F	F	T	F	T	T	
8	F	F	F	F			

• با توجه به جدول درستی، داریم:

کلاز اصلی (دترمینان)	زوج سطرها
a	(3,7)
b	(5,7)
c	$\{1,3,5\} \times \{2,4,6\} = \{(1,2), (1,4), (1,6), (3,2), (3,4), (3,6), (5,2), (5,4), (5,6)\}$

برای a=F

برای a=T

پس داریم: $TR_{GACC}=\{(3,7), (5,7), (1,4)\}$

یکی از ۹ حالت به دست آمده به دلخواه

یک مثال کامل – پوشش RACC (ت)

- می توانیم از GACC استفاده کنیم و زوج سطرهایی را که در آنها کلازهای فرعی ثابت نمانده اند را حذف کنیم.

ردیف	a	b	c	p	p_a	p_b	p_c
1	T	T	T	T			T
2	T	T	F	F			T
3	T	F	T	T	T		T
4	T	F	F	F			T
5	F	T	T	T		T	T
6	F	T	F	F			T
7	F	F	T	F	T	T	
8	F	F	F	F			

2 3 1

برای a

کلاز اصلی (دترمینان)	زوج سطرها
a	(3,7)
b	(5,7)
c	{(1,2), ~~(1,4)~~, ~~(1,6)~~, ~~(3,2)~~, (3,4), ~~(3,6)~~, ~~(5,2)~~, ~~(5,4)~~, (5,6)}

پس داریم: $TR_{RACC}=\{(3,7), (5,7), (1,2)\}$

یکی از ۳ حالت به دست آمده به دلخواه

یک مثال کامل – پوشش CACC (ث)

• می توانیم از GACC استفاده کنیم و زوج سطرهایی را انتخاب کنیم که حتما ارزش گزاره هایشان تغییر کرده باشد.

ردیف	a	b	c	p	p_a	p_b	p_c
1	T	T	T	T			T
2	T	T	F	F			T
3	T	F	T	T	T		T
4	T	F	F	F			T
5	F	T	T	T		T	T
6	F	T	F	F			T
7	F	F	T	F	T	T	
8	F	F	F	F			

2 3 1

برای a

کلاز اصلی (دترمینان)	زوج سطرها
a	(3,7)
b	(5,7)
c	{(1,2), (1,4), (1,6), (3,2), (3,4), (3,6), (5,2), (5,4), (5,6)}

گزاره را F میکند.

گزاره را T میکند.

برای گزاره این مثال، زوج سطرهای به دست آمده برای CACC کاملاً معادل GACC است (ولی لزوماً نیاز نیست TRهای نهایی آنها یکسان باشد). پس داریم:

TR_{CACC}={(3,7), (5,7), (3,4)}

یکی از ۹ حالت به دست آمده به دلخواه

یک مثال کامل – پوشش GICC (ج)

- برای هر کلاز مشخص می کنیم وقتی دترمینان نیست با کدام زوج سطرها p برابر با T یا برابر با F می شود.

ردیف	a	b	c	p	p_a	p_b	p_c
1	T	T	T	T			T
2	T	T	F	F			T
3	T	F	T	T	T		T
4	T	F	F	F			T
5	F	T	T	T		T	T
6	F	T	F	F			T
7	F	F	T	F	T	T	
8	F	F	F	F			

3 2 1

برای a

برای a=F

برای a=T

ناممکن!

کلاز اصلی (غیردترمینان)	گزاره	زوج سطرها
a	p=T	(1,5)
	p=F	{2,4} × {6,8} ={(2,6), (2,8), (4,6), (4,8)}
b	p=T	(1,3)
	p=F	{2,6} × {4,8} ={(2,4), (2,8), (6,4), (6,8)}
c	p=T	-
	p=F	(7,8)

پس داریم: TR_{GICC}={(1,5), (2,8), (1,3), ~~(2,8)~~, (7,8)}

یکی از ۴ حالت داخل جدول به دلخواه

یک مثال کامل – پوشش RICC (چ)

• می توانیم از GICC استفاده کنیم و اگر کلازهای فرعی مقادیرشان تغییر کرده آنها را از لیست نیازمندیهای آزمون حذف کنیم.

ردیف	a	b	c	p	p_a	p_b	p_c
1	T	T	T	T			T
2	T	T	F	F			T
3	T	F	T	T	T		T
4	T	F	F	F			T
5	F	T	T	T		T	T
6	F	T	F	F			T
7	F	F	T	F	T	T	
8	F	F	F	F			

3 4 2 1

برای a

کلاز اصلی (غیردترمینان)	گزاره	زوج سطرها
a	p=T	(1,5)
	p=F	{2,4} × {6,8} ={(2,6), ~~(2,8)~~, ~~(4,6)~~, (4,8)}
b	p=T	(1,3)
	p=F	{2,6} × {4,8} ={(2,4), ~~(2,8)~~, ~~(6,4)~~, (6,8)}
c	p=T	-
	p=F	(7,8)

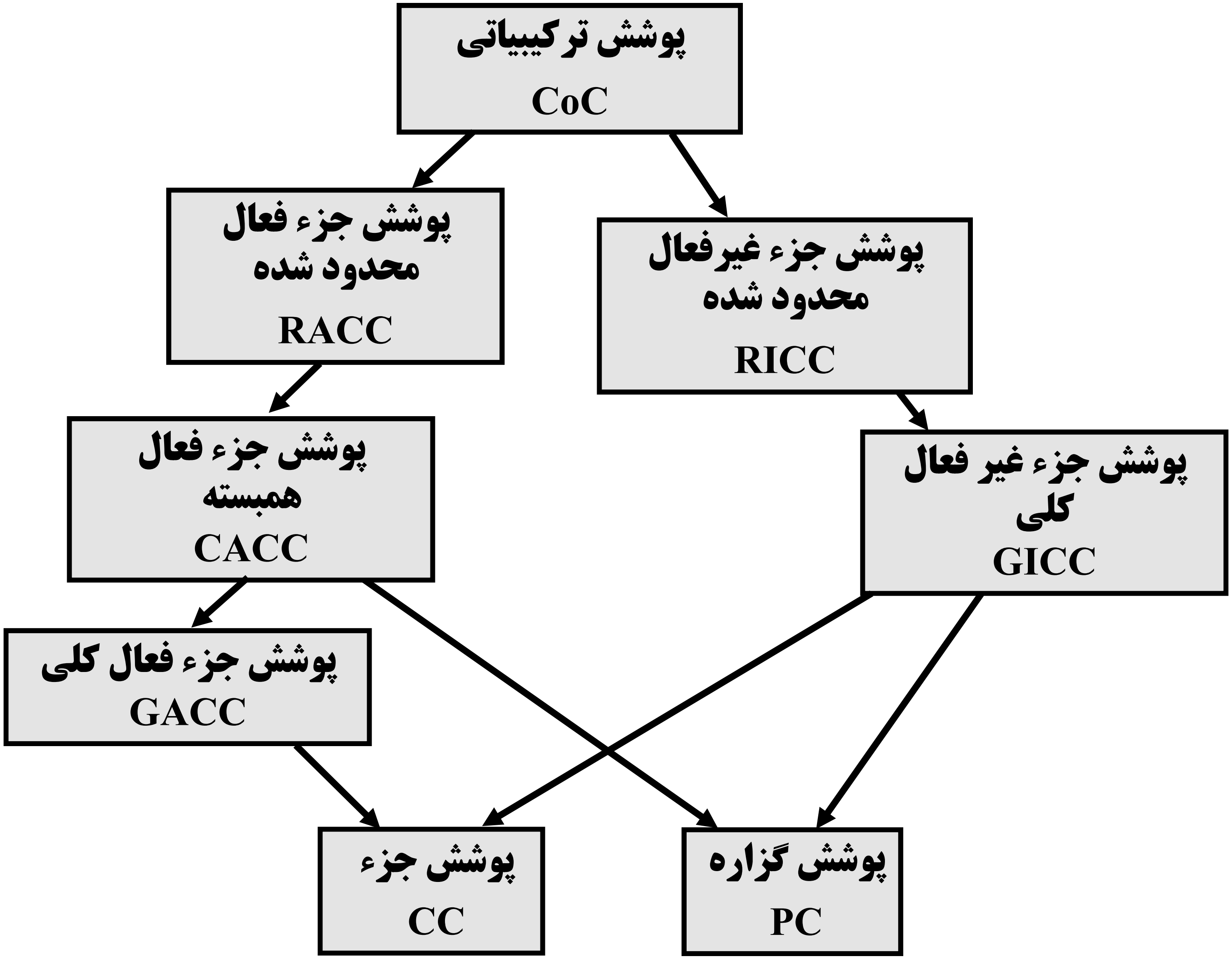
پس داریم: TR_{RICC}={(1,5), (2,6), (1,3), (2,4), (7,8)}

یکی از ۲ حالت داخل جدول به دلخواه

خلاصه معیارهای منطق

معیار	هدف	نیازمندی		شرایط	
پوشش گزاره (PC)	هر گزاره p	p=T p=F		-	
پوشش جزء (CC)	هر کلاز (جزء) c	c=T c=F		-	
پوشش جزء فعال (ACC)	هر کلاز ci در حالت فعال (دترمینان)	ci=T ci=F		GACC	-
				CACC	با تغییر ارزش ci، ارزش گزاره p تغییر کند.
				RACC	با تغییر ارزش ci، ارزش سایر کلازها (cj ها) ثابت بماند.
پوشش جزء غیرفعال (ICC)	هر کلاز ci در حالت غیرفعال (غیردترمینان)	برای p=T	ci=T ci=F	GICC	-
		برای p=F	ci=T ci=F	RACC	با تغییر ارزش ci، ارزش سایر کلازها (cj ها) ثابت بماند.

رابطه در بر داشتن میان معیارهای پوشش منطق



خلاصه فصل

- گزاره ها معمولاً بسیار ساده هستند؛ در عمل، بیشتر آنها کمتر از ۳ کلاز دارند.
  - در واقع،بیشتر گزاره ها فقط یک کلاز دارند!
  - با فقط یک کلاز، پوشش گزاره (PC) کافی است.
  - با دو یا سه کلاز، پوشش ترکیبیاتی CoC قابل انجام است.
  - مزایای معیارهای ACC و ICC برای گزاره های بزرگتر است.

- نرم افزارهای کنترل و نرم افزارهای حساس به ایمنی، معمولاً تعداد زیادی گزاره پیچیده با تعداد کلازهای زیاد دارند.
  - هوانوردی امریکا به همین دلیل پوشش منطق (و مشخصاً RACC) را برای نرم افزارهای هوانوردی اجباری کرده است.

پایان جلسه ششم