Homework #2

2022. 2학기. 인공지능

제출일: 2022/10/18

1. (Knowledge Representation) 시맨틱 웹(Semantic Web)과 온톨로지(Ontology)의 개념을 인공지능과 관련하여 설명하고, 온톨로지를 표현하기 위한 언어와 도구들에 대해 간략히 기술하시오. 인터넷 등에서 자료를 찾아 정리하되, 2 페이지 이내로 작성하시오.

solution
Semantic Web and Ontology
✓ The Semantic Web is an extension of the World Wide Web through standards set by the W3C(World Wide Web Consortium) ✓ The goal of the Semantic Web is to make Internet data machine-readable, by representing information more meaningfully for humans and computers alike ✓ To enable the encoding of semantics with the data, the ontology is defined ✓ Ontology describes concepts, relationships between entities, and categories of things. These embedded semantics offer reasoning over data and operating with heterogeneous data sources ✓ The goal of Ontology is to achieve a common and shared knowledge that can be transmitted between people and between application systems, and to facilitate the sharing of knowledge and enabling of processing between programs, services, agents or organisations across a given domain
Ontology language and tools
✓ To enable the encoding of semantics with the data, technologies such as Resource Description Framework (RDF), Web Ontology Language (OWL) are used ✓ RDF is a standard model for data interchange on the Web ✓ RDF extends the linking structure of the Web to use URIs to name the relationship between things as well as the two ends of the link (usually referred to as a “triple”) ✓ This linking structure forms a directed, labeled graph, where the edges represent the named link between two resources, represented by the graph nodes. ✓ OWL is a Semantic Web language designed to represent rich and complex knowledge about things, groups of things, and relations between things ✓ OWL is a computational logic-based language such that knowledge expressed in OWL can be exploited by computer programs, e.g., to verify the consistency of that knowledge or to make implicit knowledge explicit ✓ OWL documents, known as ontologies, can be published in the World Wide Web and may refer to or be referred from other OWL ontologies ✓ OWL is part of the W3C’s Semantic Web technology stack, which includes RDF, RDFS, SPARQL, etc. ✓ W3C announced the new version of OWL in 2009 called OWL 2, and semantic browser and editors such as Protégé, semantic reasoners such as FaCT++, RDFox were developed

2. (Predicate Logic) 다음의 문장들을 보고 답하시오.

S1. "고열이 있고 고위험군이면 Tamiflu를 처방한다“
S2. “고열이 있고 저위험군이면 Aspirin을 처방한다”
S3. “나이가 많고 고혈압이면 고위험군이다”
S4. “심장병이 있으면 고위험군이다”
S5. “나이가 젊으면 저위험군이다”
S6. “Kim 씨는 젊다”
S7. “Kim 씨는 고열이 있다”
S8. “Kim 씨는 고혈압이 있다”

1) S1 ~ S8을 모두 predicate logic sentence로 나타내시오. 다음과 같은 술어를 쓰시오.

X는 나이가 젊다/많다:	age(X, young / old)
X는 고열이 있다/없다:	fever(X, high / none)
X는 고혈압이 있다/없다:	bp(X, high / low)
X는 심장병이 있다:	has(X, heartDisease)
X는 위장병이 있다:	has(X, stomachDisorder)
X는 고/저 위험군이다:	group(X, highRisk / lowRisk)
X에게 Y를 처방한다:	prescript(X, Y)

solution
S1 : ∀X fever(X, high)∧group(X, highRisk) ⇒ prescript(X, tamiflu)
S2 : ∀X fever(X, high)∧group(X, lowRisk) ⇒ prescript(X, aspirin)
S3 : ∀X age(X, old)∧bp(X, high) ⇒ group(X, highRisk)
S4 : ∀X has(X, heartDisease) ⇒ group(X, highRisk)
S5 : ∀X age(X, young) ⇒ group(X, lowRisk)
S6 : age(kim, young)
S7 : fever(kim,high)
S8 : bp(kim, high)

2) S1 ~ S8로부터 forward chaining으로 다음의 문장을 추론하는 과정을 보이시오.

S9: prescript(kim, aspirin)

solution	
1) Unify S6 , S5 : {kim/X} → age(kim, young) ⇒ group(kim, lowRisk)	S10
2) Apply G.M.P S6, S10 → group(kim, lowRisk)	S11
3) Unify S7, S11, S2 : {kim/X} → fever(kim, high)∧group(kim, lowRisk) ⇒ prescript(kim, aspirin)	S12
4) Apply G.M.P S7, S11, S12 → prescript(kim, aspirin)	

3) 다음의 문장을 backward chaining으로 추론하여 처방할 약을 결정하는 과정을 보이시오.

S9: prescript(kim, X)

<table>
<tr><th colspan="2">solution</th></tr>
<tr><td colspan="2">사용하는 sentence 및 변수명 구분
S2 : fever(X_1, high)∧group(X_1, lowRisk) ⇒ prescript(X_1, aspirin)
S5 : age(X_2, young) ⇒ group(X_2, lowRisk)
S6 : age(kim, young)
S7 : fever(kim,high)
S9: prescript(kim, X)</td></tr>
<tr><td>1) Unify S9, S2 : {kim/X_1, aspirin/X}
→ subgoal : fever(kim, high)∧group(kim, lowRisk)</td><td>S10</td></tr>
<tr><td>2) Apply A.E. S10:
→ subgoal : fever(kim, high) → True
→ subgoal : group(kim, lowRisk)</td><td>S11
S12</td></tr>
<tr><td>3) Unify S12, S5 : {kim/X_2}
→ subgoal : age(kim, young) → True</td><td>S13</td></tr>
<tr><td>4) subgoal S13 is True → subgoal S12 is True</td><td></td></tr>
<tr><td>5) subgoal S11 is True ∧ subgoal S12 is True → subgoal S10 is True</td><td></td></tr>
<tr><td>6) subgoal S10 is True → subgoal S9 is True</td><td></td></tr>
<tr><td>∴ X = aspirin</td><td></td></tr>
</table>

3. (Prolog) 문제 2를 다음과 같이 prolog 프로그램으로 작성하여 실행하시오.

1) S1 ~ S9를 prolog 프로그램으로 작성하시오.

solution
S1 : prescript(X,tamiflu) :- fever(X,high), group(X,highRisk).
S2 : prescript(X,aspirin) :- fever(X,high), group(X,lowRisk).
S3 : group(X,highRisk) :- age(X,old), bp(X,high).
S4 : group(X,highRisk) :- has(X,heartDisease).
S5 : group(X,lowRisk) :- age(X,young).
S6 : age(kim,young).
S7 : fever(kim,high).
S8 : bp(kim,high).

2) 'prescript(kim, X)'를 question으로 주었을 때 실행 과정을 한 단계씩 보이시오.

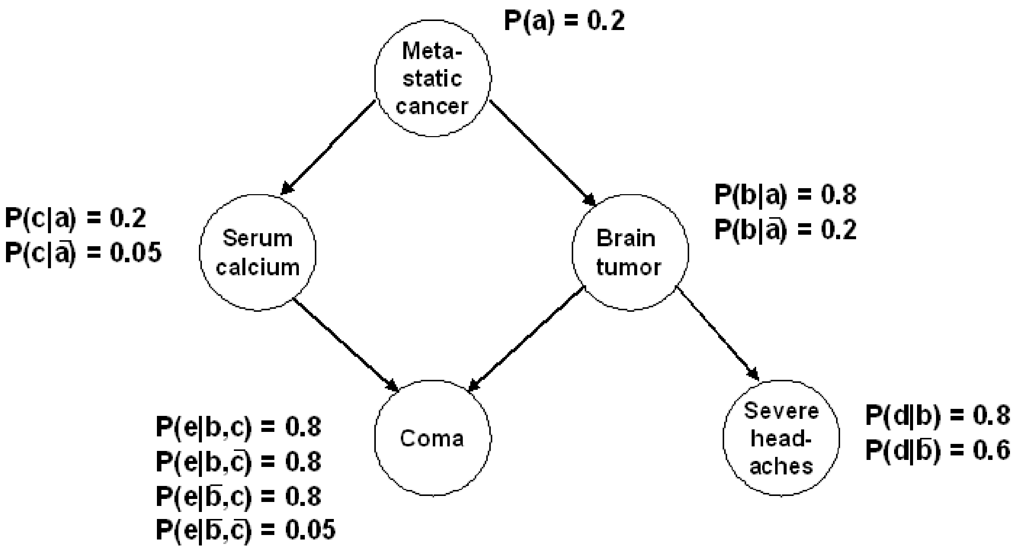
(code 및 trace 결과 제출)

solution

```
?-
% c:/Users/chltn/Documents/Prolog/predicate.pl
?- trace.
true.

[trace]  ?- prescript(kim,X).
   Call: (10) prescript(kim, _7872) ? creep
   Call: (11) fever(kim, high) ? creep
   Exit: (11) fever(kim, high) ? creep
   Call: (11) group(kim, highRisk) ? creep
   Call: (12) age(kim, old) ? creep
   Fail: (12) age(kim, old) ? creep
   Redo: (11) group(kim, highRisk) ? creep
   Fail: (11) group(kim, highRisk) ? creep
   Redo: (10) prescript(kim, _7872) ? creep
   Call: (11) fever(kim, high) ? creep
   Exit: (11) fever(kim, high) ? creep
   Call: (11) group(kim, lowRisk) ? creep
   Call: (12) age(kim, young) ? creep
   Exit: (12) age(kim, young) ? creep
   Exit: (11) group(kim, lowRisk) ? creep
   Exit: (10) prescript(kim, aspirin) ? creep
X = aspirin.
```

4. (Bayesian Networks) 아래 그림은 brain tumor 환자와 관련된 확률 추론을 하기 위해 작성된 Bayesian network의 예이다. 각 노드는 환자의 특정한 상태를 표현하며, 상태값은 모두 T 또는 F 이다. 화살표는 그러한 상태들 사이의 인과관계를 나타낸다. 예를 들어 brain tumor와 serum calcium problem은 coma를 일으키는 직접적인 원인이 된다. 또한 brain tumor는 severe headache와 coma를 일으킨다.



1) 이 network과 주어진 확률들로부터 어떤 환자가 severe headache가 있고, comma와 serum calcium problem이 있다고 할 때, 이 환자가 meta-static cancer일 확률은 얼마인지 추론해보시오.

solution

구하고자 하는 값 : P(a=True|c=True, d=True, e=True)

1. remove c=False, d=False, e=False

a	meta-static cancer
b	brain tumor
c	serum calcium
d	severe head-aches
e	coma

P(a=true)	0.2
P(a=false)	0.8

p(c¦a)	a=true	a=false
c=true	0.2	0.05
~~c=false~~	~~0.8~~	~~0.95~~

p(b¦a)	a=true	a=false
b=true	0.8	0.2
b=false	0.2	0.8

p(e¦b,c)	b=true		b=false	
	c=true	~~c=false~~	c=true	~~c=false~~
e=true	0.8	~~0.8~~	0.8	~~0.05~~
~~e=false~~	~~0.2~~	~~0.2~~	~~0.2~~	~~0.95~~

p(d¦b)	b=true	b=false
d=true	0.8	0.6
~~d=false~~	~~0.2~~	~~0.4~~

2. αP(a,b|c,d,e)

	b=True	b=False
a=true	0.8*0.8*0.8*0.2*0.2=0.02048	0.6*0.8*0.2*0.2*0.2=0.00384
a=false	0.8*0.8*0.2*0.05*0.8=0.00512	0.6*0.8*0.8*0.05*0.8=0.01536

3. αP(a|c,d,e) - sum over b

a=true	0.02432
a=false	0.02048

4. nomalize

a=true	0.5429
a=false	0.4571

∴ P(a=True|c=True, d=True, e=True) = 약 0.54

or

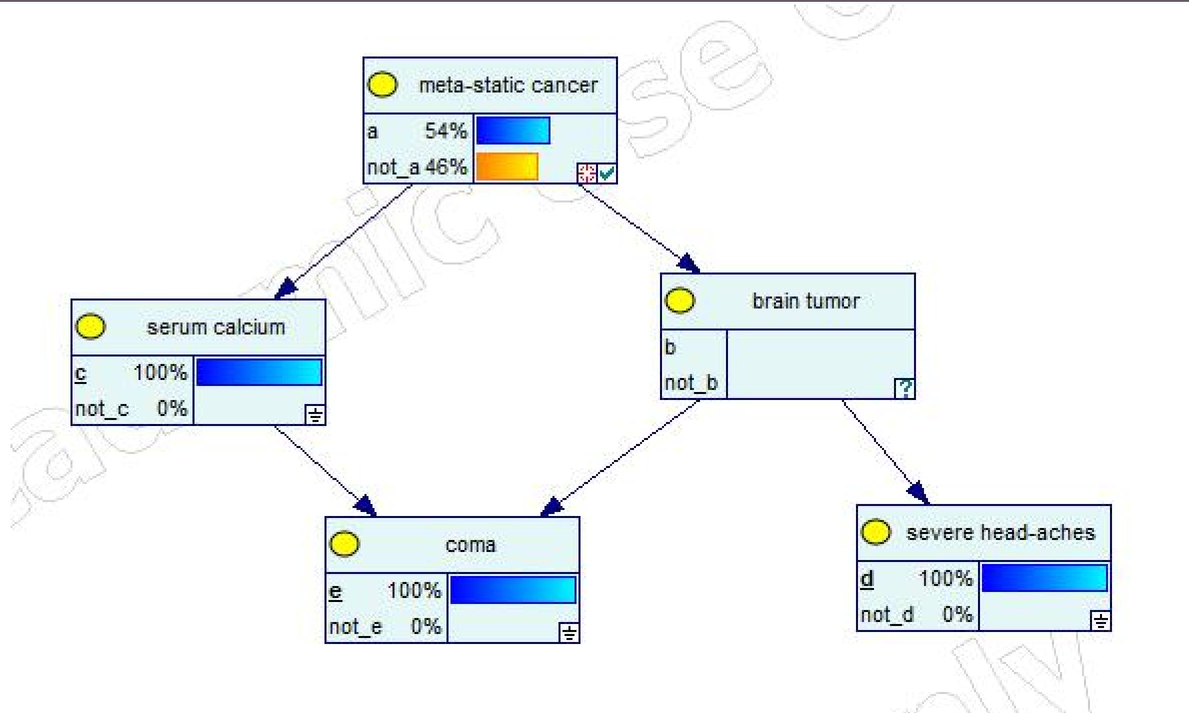
$$
\begin{aligned}
P(a=true\,|\,c=true, d=true, e=true) &= \alpha P(a=true, c=true, d=true, e=true) \\
&= \alpha \sum_b P(a=true, c=true, d=true, e=true, b) \\
&= \alpha \sum_b P(d=true|b)P(e=true|b,c=true)P(b|a=true)P(c=true|a=true)P(a=true) \\
&= \alpha P(c=true|a=true)P(a=true) \sum_b P(d=true|b)P(e=true|b,c=true)P(b|a=true) \\
&= \alpha \times 0.2 \times 0.2 \times [0.8 \times 0.8 \times 0.8 + 0.6 \times 0.8 \times 0.2] = 0.02432\alpha
\end{aligned}
$$

$$
\begin{aligned}
P(a=false\,|\,c=true, d=true, e=true) &= \alpha P(a=false, c=true, d=true, e=true) \\
&= \alpha \sum_b P(a=false, c=true, d=true, e=true, b) \\
&= \alpha \sum_b P(d=true|b)P(e=true|b,c=true)P(b|a=false)P(c=true|a=false)P(a=false) \\
&= \alpha P(c=true|a=false)P(a=false) \sum_b P(d=true|b)P(e=true|b,c=true)P(b|a=false) \\
&= \alpha \times 0.05 \times 0.8 \times [0.8 \times 0.8 \times 0.2 + 0.6 \times 0.8 \times 0.8] = 0.02048\alpha
\end{aligned}
$$

∴ P(a=True|=True, d=True, e=True) = 0.02432/(0.02432+0.02048) = 약 0.54

2) GeNIe software (www.bayesfusion.com)를 이용하여 이 network를 작성하고 1)의 추론을 진행하여 결과를 확인하시오. (screenshot 제출)

solution



5. (Hidden Markov Model) Hidden states S1, S2, S3의 transition probability와 각 state에서의 observation R, G, B의 output probability가 다음과 같다고 하자.

P(S1) = P(S2) = P(S3) = 1/3

	S1	S2	S3
S1	0.6	0.2	0.2
S2	0.1	0.3	0.6
S3	0.3	0.1	0.6

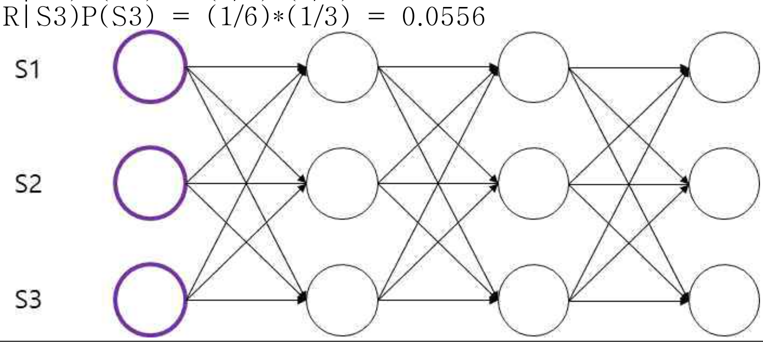
	R	G	B
S1	3/6	2/6	1/6
S2	1/6	3/6	2/6
S3	1/6	1/6	4/6

1) 아래와 같은 observation이 주어졌을 때, 가장 확률이 높은 hidden state sequence를 Viterbi algorithm을 사용하여 예측하시오. 계산 과정을 보이시오.

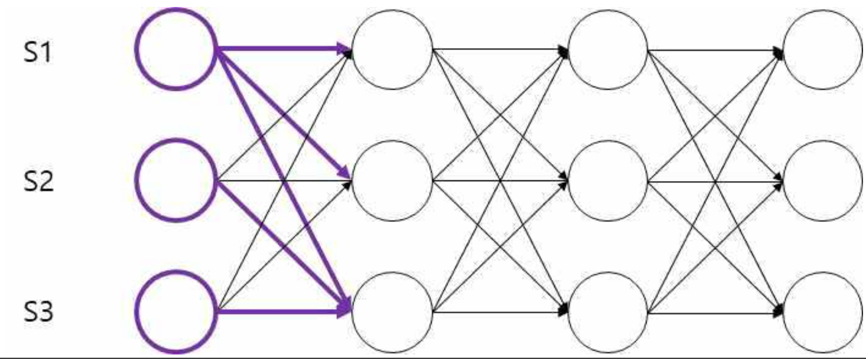
Observation: (R, R, G, B)

solution

```
score1_S1 = P(R|S1)P(S1) = (3/6)*(1/3) = 0.1667
score1_S2 = P(R|S2)P(S2) = (1/6)*(1/3) = 0.0556
score1_S3 = P(R|S3)P(S3) = (1/6)*(1/3) = 0.0556
```



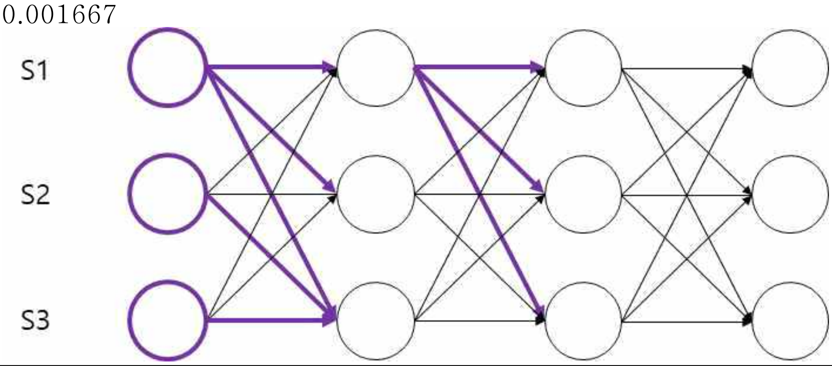

```
score2_S1 = max P(R|S1)P(S1|X1)score(X1)
          = max{ P(R|S1)P(S1|S1) score1_S1 = (3/6)*0.6*0.1667 = 0.05001,
                 P(R|S1)P(S1|S2) score1_S2 = (3/6)*0.1*0.0556 = 0.00278,
                 P(R|S1)P(S1|S3) score1_S3 = (3/6)*0.3*0.0556 = 0.00834 }
          = 0.05001
score2_S2 = max P(R|S2)P(S2|X1)score(X1)
          = max{ P(R|S2)P(S2|S1) score1_S1 = (1/6)*0.2*0.1667 = 0.00556,
                 P(R|S2)P(S2|S2) score1_S2 = (1/6)*0.3*0.0556 = 0.00278,
                 P(R|S2)P(S2|S3) score1_S3 = (1/6)*0.1*0.0556 = 0.00093 }
          = 0.00556
score2_S3 = max P(R|S3)P(S3|X1)score(X1)
          = max{ P(R|S3)P(S3|S1) score1_S1 = (1/6)*0.2*0.1667 = 0.00556,
                 P(R|S3)P(S3|S2) score1_S2 = (1/6)*0.6*0.0556 = 0.00556,
                 P(R|S3)P(S3|S3) score1_S3 = (1/6)*0.6*0.0556 = 0.00556}
          = 0.00556
```




```
score3_S1 = max P(G|S1)P(S1|X2)score(X2)
          = max{ P(G|S1)P(S1|S1) score2_S1 = (2/6)*0.6*0.05001 = 0.010002,
                 P(G|S1)P(S1|S2) score2_S2 = (2/6)*0.1*0.00556 = 0.000185,
                 P(G|S1)P(S1|S3) score2_S3 = (2/6)*0.3*0.00556 = 0.000556}
          = 0.010002
score3_S2 = max P(G|S2)P(S2|X2)score(X2)
```

= max{ P(G|S2)P(S2|S1) score2_S1 = (3/6)*0.2*0.05001 = 0.005001,
P(G|S2)P(S2|S2) score2_S2 = (3/6)*0.3*0.00556 = 0.000834,
P(G|S2)P(S2|S3) score2_S3 = (3/6)*0.1*0.00556 = 0.000278}
= 0.005001

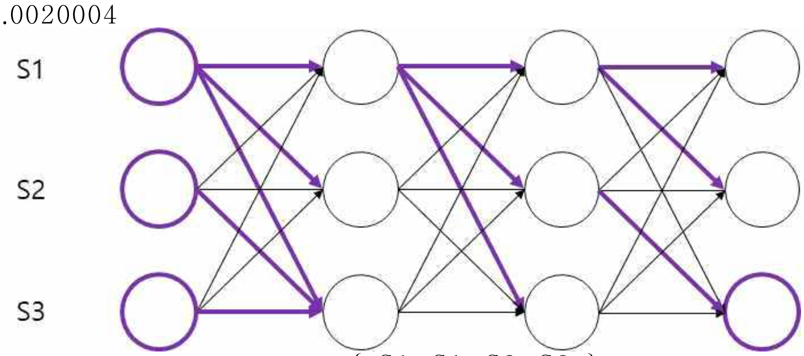
score3_S3 = max P(G|S3)P(S3|X2)score(X2)
= max{ P(G|S3)P(S3|S1) score2_S1 = (1/6)*0.2*0.05001 = 0.001667,
P(G|S3)P(S3|S2) score2_S2 = (1/6)*0.6*0.00556 = 0.000556,
P(G|S3)P(S3|S3) score2_S3 = (1/6)*0.6*0.00556 = 0.000556}
= 0.001667



score4_S1 = max P(B|S1)P(S1|X3)score(X3)
= max{ P(G|S1)P(S1|S1) score3_S1 = (1/6)*0.6*0.010002 = 0.0010002,
P(G|S1)P(S1|S2) score3_S2 = (1/6)*0.1*0.005001 = 0.00008335,
P(G|S1)P(S1|S3) score3_S3 = (1/6)*0.3*0.001667 = 0.00008335 }
= 0.0010002

score4_S2 = max P(B|S2)P(S2|X3)score(X3)
= max{ P(G|S2)P(S2|S1) score3_S1 = (2/6)*0.2*0.010002 = 0.0006668,
P(G|S2)P(S2|S2) score3_S2 = (2/6)*0.3*0.005001 = 0.0005001,
P(G|S2)P(S2|S3) score3_S3 = (2/6)*0.1*0.001667 = 0.0000556 }
= 0.0006668

score4_S3 = max P(B|S3)P(S3|X3)score(X3)
= max{ P(B|S3)P(S3|S1) score3_S1 = (4/6)*0.2*0.010002 = 0.0013336,
P(B|S3)P(S3|S2) score3_S2 = (4/6)*0.6*0.005001 = 0.0020004,
P(B|S3)P(S3|S3) score3_S3 = (4/6)*0.6*0.001667 = 0.0006668 }
= 0.0020004



sequence : { S1, S1, S2, S3 }