

카이스트 전산학과 대학원 입시 기출 문제 모음

2014.5

1. Programming Language / Compiler

- 자바에서 public, protected, private 키워드가 있는데 아무것도 안 쓸 경우 default로 적용되는 범위는?
- Static typing과 dynamic typing의 차이점은? C++/JAVA은 어떤 typing을 사용하는가?
- C++에서 서로 다른 타입의 오브젝트를 가리키는 포인터를 사용할 수 있나?
- 전역 변수 사용의 장단점은?
- 프로그램 실행 전 컴파일 시에 타입과 같은 프로그램의 정적 성질을 검사하는 프로그래밍 언어와 그렇지 않은 프로그래밍 언어를 비교하시오.
- 파라미터 패싱 방식에는 eager evaluation 방식과 lazy evaluation 방식이 있다. 두 방식의 차이점을 비교 설명하세요. call-by-value와 call-by-name 파라미터 패싱 방식은 각각 어느 방식에 속하는지 구분하세요.
- 언어를 분류하는 데 있어서는 해당 언어를 생성하는 Production 룰의 형태에 가해지는 제약 사항을 가지고 분류하는 것이 일반적이다. Context Sensitive Language, Context Free Language, Regular Language를 생성하는 데 사용되는 Production 룰의 형태에 대해서 그 차이점을 비교 설명하세요.
- Let L be a regular language. Prove that R(L), strings in L reversed, is also a regular language.
- LR 파싱이란? Parsing Table의 역할은? 핸들이란? 핸들의 가장 중요한 특징은?
- 파싱에서 bottom-up 방식과 top-down 방식의 차이점은?
- Recursive-descent 방식에서 백트래킹이 일어나지 않게 하려면 어떻게 해야 하는가?
- A grammar G generating language L is defined by:
- Consider the following grammar, with start symbol E:

E -> E * E

| E / E

| E + E

| E - E

| (E)

| a | b | c | d | e | f x | y | z

The following strings are legal derivations from this grammar:

- I. $a * b + c$
- II. $(a - b) * c$
- III. $a / (b - c)$

Which of the above are rightmost sentential forms?

- A relation on the integer 0 through 4 is defined by:

$$R = \{(x,y) \mid x+y \leq 2x\}$$

Which of the properties listed below applies to this relation? Why?

- I. Transitivity
- II. Symmetry
- III. Reflexivity

- Grammar G is defined by:

$$G = (\{x,y,z\}, \{S,W,X,Y,Z\}, P, S)$$

Where the members of P are:

$$S \rightarrow WZ$$

$$W \rightarrow X \mid Y$$

$$X \rightarrow x \mid xY$$

$$Y \rightarrow y \mid yY$$

$$Z \rightarrow z \mid zZ$$

Which of the following regular expressions corresponds to this grammar?

- (A) $xx^* \mid yy^* \mid zz^*$
- (B) $(xx^* \mid yy^*).zz^*$

(C) $xx^*(yy^* \mid zz^*)$

(D) $(xx \mid yy)^*.zz^*$

(E) $x^*.yy^*.xx^*$

- A computer system stores floating-point numbers with a 16-bit mantissa and an 8-bit exponent, each in two's complement. The smallest and largest positive values which can be stored are

(A) 1×10^{-128} and $2^{15} \times 10^{128}$

(B) 1×10^{-256} and $2^{15} \times 10^{255}$

(C) 1×10^{-128} and $2^{15} \times 10^{127}$

(D) 1×10^{-128} and $(2^{15}-1) \times 10^{127}$

(E) 1×10^{-256} and $(2^{15}-1) \times 10^{255}$

- A grammar G generating language L is defined by:

$$G = (\{x,y\}, \{S,X,Y\}, P, S)$$

where the elements of P are:

$$S \rightarrow xY$$

$$S \rightarrow yX$$

$$X \rightarrow xZ$$

$$X \rightarrow x$$

$$Y \rightarrow y$$

$$Z \rightarrow z$$

Is the language L generated by G most accurately said to be:

- A. Chomsky type 0
- B. Chomsky type 1 (context sensitive)
- C. Chomsky type 2 (context free)
- D. Chomsky type 3 (regular)

E. None of the above

- Consider the following program fragment:

```
read(a, b)
```

```
c := 3.0 * a + b ;
```

```
if c = 0 then a :=1 else a := 1.0/c +1.0/b ;
```

This program fragment will fault if given certain values for "a" and "b." Which is the weakest of the supplied conditions (least restrictive or *smallest*) which, if applied to the data, will prevent failure?

- A. $b > 0$
- B. $a > 0$ and $b > 0$
- C. $a \neq -b/3$
- D. $b \neq 0$
- E. $3.0*a \neq 0$ and $b \neq 0$

- Grammar G is defined by:

$G = (\{x,y,z\}, \{S,W,X,Y,Z\}, P, S)$

Where the members of P are:

$S \rightarrow WZ$

$W \rightarrow X \mid Y$

$X \rightarrow x \mid xY$

$Y \rightarrow y \mid yY$

$Z \rightarrow z \mid zZ$

- A. Show some examples that are generated by this grammar.
- B. Write the regular expression corresponds to this grammar?

- A relation over the set $s = \{x,y,z\}$ is defined by:

$\{(x,x), (x,y), (y,x), (x,z), (y,z), (y,y), (z,z)\}$

What properties hold for this relation?

- I. Symmetry
- II. Reflexivity
- III. Antisymmetry
- IV. Irreflexivity
- V. Transitivity

■ Grammar G is defined by:

$G = (\{a,b\}, \{S,A,B\}, P, S)$

Where the members of P are:

$S \rightarrow AB \mid AS$

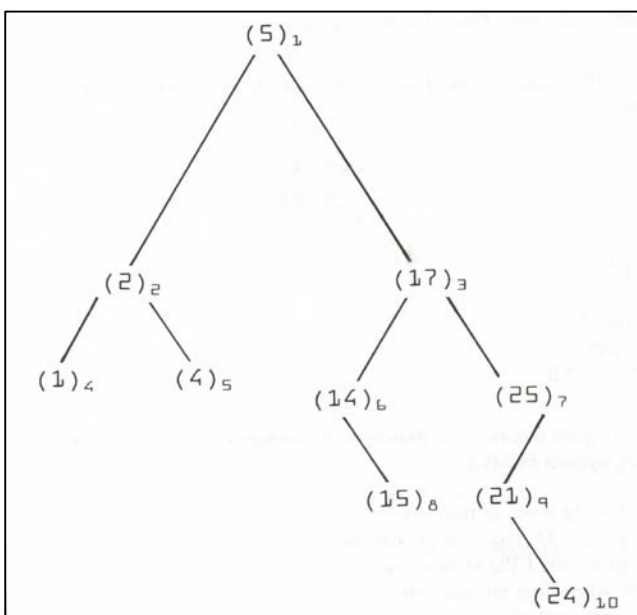
$A \rightarrow a \mid aA$

$B \rightarrow b$

A. Show some examples that are generated by this grammar.

B. Write the regular expression corresponds to this grammar?

■ A binary tree is to be used to sort positive integers. The partial tree shown below is formed.



The next number in the file is **13**; where should this be placed such that in order traversal of the tree yields the sorted file?

- A grammar G generating language L is defined by:

$G = (\{x,y\}, \{S,A,B,C\}, P, S)$

where the elements of P are:

$S \rightarrow ABA$

$AB \rightarrow AC$

$CB \rightarrow BBC$

$CA \rightarrow BBA$

$A \rightarrow a \mid E$

$B \rightarrow b$

Is the language L generated by G most accurately said to be:

- A. Chomsky type 0
- B. Chomsky type 1 (context sensitive)
- C. Chomsky type 2 (context free)
- D. Chomsky type 3 (regular)
- E. None of the above

- Consider the following program fragment and its assertions:

ASSERT INITIAL (B > 0) AND (C > 0)

If B > C

THEN A := B / C

ELSE A := C / B

ENDIF;

ASSERT FINAL (A > 1)

There are *two* paths through this code; which of the two is taken depends upon the *predicate* "B > C". Which of the following statements is true?

- A. The program fragment is consistent with its assertions along *both* path.
- B. The program fragment is inconsistent with its assertions along *both* path.
- C. The program fragment is consistent with its assertions along the *predicate = false* path only.
- D. The program fragment is consistent with its assertions along the *predicate = true* path only.
- E. The question of consistency is undecidable until the values of B and C are known.

■ Grammar G is defined by:

$$G = (\{a,b\}, \{S,A,B\}, P, S)$$

where P is the set:

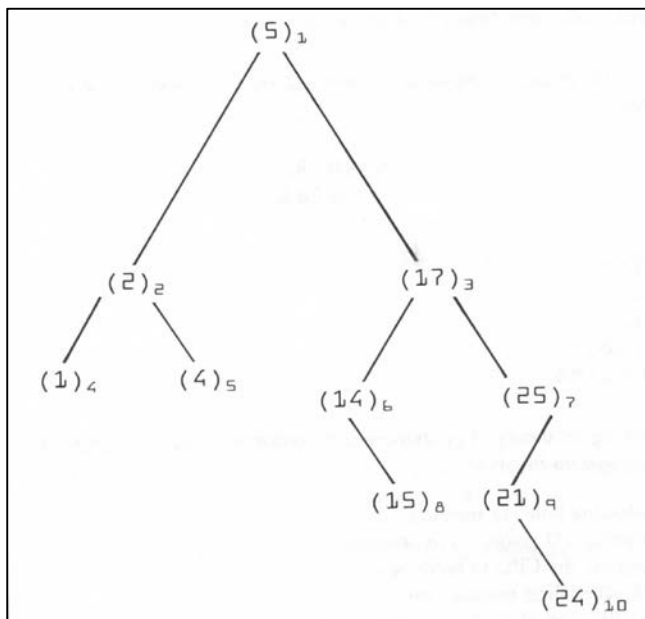
$$S \rightarrow AB \mid AS$$

$$A \rightarrow a \mid aA$$

$$B \rightarrow b \mid bb$$

- A. Draw a parse tree for the string "aaabb".
- B. What type is the language L(G) most accurately? Is it regular, context free, or context sensitive?

■ The binary tree below is traversed in postorder. In what order are the nodes visited?



Write the program to visit a binary tree T in postorder and print the value of nodes in your favorite programming language.

2. Architecture

- 캐시가 필요한 이유는? Cache hit ratio 에 대해 설명하시오.
- 메모리 접근하는데 x 사이클이 걸리고 캐시에 접근하는데 y 사이클이 걸리며 캐시 hit rate 가 h %일 때 effective access time은?
- 페이지 폴트는 언제 발생하는가? 페이지 폴트 비율과 cache miss 비율 중 큰 것은? 그 이유는?
- 캐시 메모리와 메인 메모리의 주소 지정 방식의 차이점이 무엇인가?
- 캐시를 구성하는 컴포넌트에 무엇이 있는가? 각 컴포넌트는 어떤 역할을 하는가?
- 캐시에서 태그 매칭이 무엇이고 왜 필요한가?
- 캐시에서 블록(라인) 크기를 크게 했을 때와 작게 했을 때 어떤 장단점이 있을까?
- Direct-mapped cache와 set-associative cache의 장단점은 무엇인가?
- Write-through cache와 write-back cache에 대해서 설명해 보시오.
- Write-through cache는 write buffer를 보통 사용하는데 이의 역할은 무엇인가?
- 코어가 10개 있는 cpu에서 프로세스 하나를 균등하게 처리하면 이상적인 경우, 시간이 얼마나 줄어드는가? 그런데 프로세스에서 분할되지 않는 20%의 작업이 있다고 하면, 시간이 얼마나 줄어드는가? 이제 코어가 수백개, 혹은 무한개 있다고 하면 얼마나 줄어드는가?
- Cache에서 사용하는 replacement policy에는 어떤 것이 있는가?

3. OS

- Kernel을 필요에 의해 어느정도 수정했다고 하자. 이 kernel이 제대로 작동하는지를 알기 위해서 어떤 test를 해야겠나?
- Sequential program과 multithread program에서 error detection에 대한 차이점에는 어떤 것이 있나?
- 어떤 C program으로 작성되어 수행중인 process가 있다고 가정하자. C언어에서는 직접적으로 주소를 변수에게 지정해 줄 수 있다. 만약 주소 1, 2, 3, 4에 변수를 잡아서 어떤 일을 수행하는 process라고 하자. 이 process를 한 시스템에 동시에 두번 수행시켰다. 그랬을 때 한 프로세스가 1, 2, 3, 4 주소에 있는 변수를 바꿨을 때 다른 프로세스의 변수들에도 영향을 끼치는가?
- Thread와 process의 차이는 무엇인가?
- IPC가 무엇인가?
- Logical address와 physical address의 차이는 무엇인가?
- Logical address를 physical address로 바꾸어주는 hw가 무엇인가?
- Interrupt란 무엇인가? interrupt를 두 종류로 나눈다면 어떻게 되는가? (software interrupt/hardware interrupt) 두 interrupt의 차이는 무엇인가?-두 interrupt의 handler를 서로 구분해서 구현해야 하는 것이 좋은가, 아니면도 상관 없는가?
- C로 숫자로 직접 입력된 주소를 참조하는 프로그램을 짜서 컴파일한 후, 두 개를 실행시켰다고 하자. 그러면 이 두 프로세스는 물리적으로 같은 곳을 참조하나? 만약 아니라면, 어떻게 서로 다른 물리적 공간을 참조할 수 있나?
- 프로세스는 가상 메모리 주소로 어떻게 실제 메모리 주소를 찾아가죠?
- Paging이 무엇인가? paging을 할 때 어떻게 실제 메모리 주소에 데이터를 전송하는가? page table에는 어떤 항목이 저장되는가? page table은 어디에 저장되는가? page table이 메모리에 저장되면, paging을 할 때 메모리를 두 번 참조해야 되는데, 좀 더 빠르게 하는 방법은 없는가? TLB에는 어떤 항목이 저장되는가?
- Thread와 Process의 차이는? Thread끼리 context switching 하는 과정에 대해 설명하십시오. 같은 프로세스 내부의 thread들끼리 전환되는 것과 다른 프로세스간의 thread끼리 전환되는 것이 어떻게 다른가?
- Virtual Address와 Virtual memory에 대해 설명하십시오.
- 시스템 콜과 인터럽트의 차이는? 인터럽트가 걸리면 어떤 일이 일어나고, 인터럽트

처리 후에 어떻게 이전 상태로 돌아가나?

- Non-preemptive scheduling이란? Critical section이란?
- OS의 캐시 메모리의 사이즈를 구하기 위한 프로그램을 어떻게 구현하면 되는가?
- 세마포어의 개념은? 주요 연산 2가지는? 어떻게 구현해야 하는가?
- Virtual memory에서 page replacement policy에는 어떤 것이 있는가? LRU의 단점은?
- 운영 체제에서 memory management 기법중 하나로 paging이 많이 사용되고 있습니다. Paging 기법의 장단점들로는 무엇이 있으며, hierarchical paging 혹은 inverted page table은 어떤 환경에서 유리할까요?
- 요즘 많은 사람들이 각각 자신만의 스마트폰을 사용하고 있습니다. 이처럼 각 개인 스마트폰을 위하여 process system을 design하려고 합니다. 스마트폰에서 각 application이 하나의 process로 독립해서 실행하는 것이 나올까요? 아니면, 하나의 thread로 만들어져서 실행하는 것이 나올까요? 어느 쪽이 나올지 결정하고, 그 이유를 설명하세요.
- 캐시 메모리와 메인 메모리의 주소 지정 방식의 차이점이 무엇인가? s

4. DB

- Data inconsistency란 무엇인가?
- Data normalization이 무엇인가? 왜 필요한가?
- Inner Join과 Outer Join의 차이점은? Outer Join은 언제 사용하는가?
- DB와 file의 차이점은?
- Data independence란?

5. 이산 수학, 확률, 통계

- Mathematical induction이란?
- Relation의 정의는?
- $\forall a \in A, \forall b \in B$ 일 때 $aRb \leftrightarrow (a, b) \in R$ 이 뭔 뜻인지 설명하시오. 여기서 왼쪽 R 과 오른쪽의 R이 같은 것인가, 다른 것인가? 다르다면 어떻게 다르나?
- Bijection의 정의는?
- Equivalence relation이란?
- Partial order relation이란? Total order relation이란?
- Random variable이 무엇인가?
- Sample space란 무엇인가?
- Sample space가 갖는 조건이 무엇인가?
- Exponential distribution이 무엇인가?
- Poisson distribution이 무엇인가?

6. 네트워크

- TCP와 같이 protocol을 reliable하게 설계 하려면 무엇이 필요한가?
- Flow control이란 무엇인가? Congestion control과 flow control의 차이점은?
- OSI 7계층에 대해 설명하시오.
- Transport, network, link 계층이 모두 연결에 관한 것인데, 무엇이 차이가 나나요?
- Network에서 checksum을 위해 사용하는 hash function과 security에서 data integrity를 위해 사용하는 hash function이 서로 interchangeable해요?
- TCP와 UDP의 차이점은?
- Congestion이 발생했다는 것을 어떻게 할 수 있는가?
- 3-way handshaking이란?
- CSMA/CD란?
- MAC address란? IP address가 MAC address에 비해 갖는 장점은?

7. Software Engineering

- UML 에서 시퀀스 다이어그램이란 무엇인가?
- SW에서 모듈화란? 모듈화를 잘 하기 위해 중요한 것은?
- CMMI 레벨이란?
- 요구 공학에서는 어떻게 요구사항을 수집하는가?
- 테스트 기법으로 화이트 박스와 블랙 박스 테스트가 있는데, 그 둘의 차이점은? 화이트 박스 테스트가 에러가 없는 프로그램이라고 보장해 주지 않는 이유는?

8. AI / ML

- Show, using a proof or an example, that if $P(A|B,C) = P(A|C)$, then $P(A,B|C) = P(A|C)P(B|C)$
- An eigenvector of a square matrix A is a non-zero vector v that, when multiplied by A , yields the original vector multiplied by the corresponding eigenvalue λ . What is the eigenvector of $\begin{bmatrix} 0 & 1 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix}$ corresponding to the eigenvalue 3?
- In A^* search, the evaluation function is defined as follows: $f(n) = g(n) + h(n)$, where $g(n)$ is the cost so far to reach n , and $h(n)$ is the estimated cost from n to goal. We call $h(n)$ the heuristic function. Under what condition is A^* search guaranteed to find the optimal solution? If we want to design an *admissible* heuristic, what condition does $h(n)$ need to satisfy?

9. 알고리즘

- Sorting algorithm 들에는 어떤 것들이 있는가? Insertion sort, heap sort, selection sort, quick sort 중 optimal 한 것과 optimal 하지 않은 알고리즘은? 그 이유는?
- 그래프의 정의는? 트리의 정의는? 트리와 그래프의 관계는?
- Topological Sorting을 설명하시오.
- Partially ordered set이란?
- NP Complete의 정의는?
- Dynamic Programming이란?
- Priority queue란? Heap 이란?
- 오일러 사이클이란?
- Tree traversal의 3 가지 방식을 설명하시오.
- Quick sort 에 대해 설명하시오.
- Transitive closure란?
- Finite state automata란?
- Binary tree란? binary tree에서 각 node는 두 개의 children을 갖거나 혹은 leaf node 이거나 둘 중에 하나라고 할 때, non-leaf node와 leaf node 개수의 관계식은 어떻게 되는가?
- Minimum spanning tree를 구하는 알고리즘과, 그 알고리즘의 복잡도는?
- Suppose we have a straightforward $\Theta(n^2)$ algorithm for a problem. Suppose we devise a divide-and-conquer algorithm that divides an input into two inputs half as big, and takes $n \lg n$ time to divide the problem and $n \lg n$ time to combine the solutions. Is the divide-and-conquer algorithm faster or slower than the straightforward algorithm? Justify your answer.
- Let P be a problem. The worst-case time complexity of P is $O(n^2)$. The worst-case complexity of P is also $\Theta(n \lg n)$. Let A be an algorithm that solves P . Which subset of the following statements are consistent with this information about the complexity of P .
 - (a) A has worst-case time complexity $O(n^2)$
 - (b) A has worst-case time complexity $O(n)$

(c) A has worst-case time complexity $\Theta(n^2)$

(d) A has worst-case time complexity $\Theta(n^3)$

- Show how you can make Quicksort to have $O(n \lg n)$ worst-case running time.
- Consider the following divide-and-conquer algorithm. Given a graph $G = (V, E)$, we partition the set V into two sets V_1 and V_2 s.t. $|V_1|$ and $|V_2|$ differ by at most 1. Let E_1 be the set of edges that are incident only on vertices in V_1 , and let E_2 be the set of edges that are incident only on vertices in V_2 . Recursively solve a minimum-spanning-tree problem on each of the two subgraphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. Finally, select the minimum-weight edge in E that crosses the cut (V_1, V_2) , and use this edge to unite the resulting two minimum spanning trees into a single spanning tree. Either argue that the algorithm correctly computes a minimum spanning tree of G , or provide an example for which the algorithm fails.
- Let T be a minimum spanning tree of G . Then for any pair of vertices s and t , is the shortest path from s to t in G the path from s to t in T ? Justify your answer.
- Describe Dijkstra's algorithm and analyze its running time.
- Describe Floyd-Warshall algorithm and analyze its running time.
- Optimal한 정렬 알고리즘의 예를 들어보세요. 그 알고리즘의 complexity를 분석하고 optimal하다는 것을 증명해보세요.
- Dynamic programming과 Greedy algorithm에 대하여 각각이 무엇인지 설명하고, 공통점과 차이점을 이야기하세요. 구체적으로 Dynamic programming으로 풀 수 있는데 Greedy algorithm으로는 풀 수 없는 문제의 예를 들어보세요. 그 문제가 dynamic programming으로 풀 수 있다는 것을 증명해보세요.
- n 개의 숫자가 array로 주어졌습니다. 이 중에 Median을 찾는 알고리즘을 설명해보세요. 설명한 알고리즘의 complexity는 무엇입니까? Median을 찾는 optimal한 알고리즘의 complexity는 무엇입니까? optimal하다는 것을 어떻게 증명할 수 있나요? Median을 찾는 알고리즘을 이용하여 n 개의 숫자를 정렬해보세요. 이 정렬 알고리즘의 complexity는 무엇입니까? 두 가지 문제를 통해 Reduction이라는 개념을 설명해보세요.

10. 그래픽

- 여러 shading 모델들과, 차이점을 설명하시오
- Edge detection에 대해 설명하시오.
- Massive한 데이터를 그래픽으로 출력하려면, 어떻게 해야 하는가?
- 3차원의 가상공간에 놓여 있는 3차원 물체들이 2차원의 그림으로 렌더링되어 우리의 화면에 보여진다. 이를 우리는 2차원 윈도우를 통해서 상호작용을 하게 되는데 이때 간단하게는 마우스를 이용하여 3차원 공간에 있는 물체를 선택하게 된다. 이때 마우스 input은 (x,y) 2차원 좌표인데 어떻게 3차원 공간에 있는 물체를 선택(selection) 할 수 있을까? 이를 가능하게 할 수 있는 Object Selection 방법을 제시하라.
- 물체의 빠른 충돌검지를 위해서 물체를 감싸는 간단한 도형인 bounding volume을 hierarchical 하게 구성하여 사용한다. 이의 간단한 예를 만들어 얼마나 빠르게 처리를 할 수 있는지 설명하라

11. 정보 보호

- 네트워크에서 정보를 전송할 때 데이터를 보호하기 위해 암호화하는 방법에는 어떤 것이 있는가?