* 제출 관련
  + 학번.zip에 필요한 C++ source code들을 압축하여 제출 (예: 2000-00000.zip)
  + 제출 기한은 10월 30일(금) 23:59까지 입니다.
    - 딜레이는 하루에 30%씩 감점합니다. 3일이 지난 11월 2일까지만 제출 받겠습니다.
* 과제 설명
  + 강의 시간에 교수님께서 말씀드렸던 대로 구현하시면 됩니다.
  + 아래의 사항들은 강의 시간에 언급되었던 요구사항들의 주요 내용들을 정리한 것입니다.
  + 용어 정리
    - Memory array: pseudocode 상의 Memory array (tree structure가 저장됨)
  + 목표
    - 사용자가 Input으로 콘솔에 한 줄의 code를 입력하면 이를 parsing해서 Symbol table (Hash table)과 Memory array에 삽입한 다음 Output으로 symbol table과 Memory array의 내용과 입력 받은 list를 출력
  + 상세
    - 입력 code는 error가 없고 single line으로 입력된다고 가정
    - 이번 과제에서 quote(')는 무시
    - Memory array의 크기는 30으로 고정
    - Symbol의 max length = 10
    - 입력 받은 symbols는 다 소문자로 변환시킬 것
    - list를 print하기 전에 출력할 것들
      * free-list와 list 각각의 root node index 출력
        + symbol 하나만 input으로 들어올 경우 심볼에 해당하는 hash table index의 음수를 list의 root node index로 symbol에 해당하는 hash table index의 음수를 출력해 주세요. (9/16 12:30 추가)
      * Memory array와 hash table의 내용 출력
        + Memory array는 그대로 출력
        + hash table은 non-empty elements만 출력 (각 element의 hash value, symbol이 출력되야 함)
    - >: input하기 전에 나오는 기호  
      ]: output하기 전에 나오는 기호
      * 예:  
        > (define (square x) (x x))  
        ] Free list's root = ""free-list의 root node index""  
        List's root = ""Memory array의 root node index""  
          
        Memory table =  
        ""Memory table 내용""  
          
        Hash table =  
        ""hash table 내용""  
          
        (define (square x) (x x))  
          
        > …  
        ] …
    - 항상 괄호나 공백을 기준으로 symbol을 나누고 그 외에는 하나의 symbol로 취급하시면 됩니다. 예를 들어, (define 1+1 5)이 입력되면 (define 1 + 1 5)로 다섯개의 symbol이 존재하는 것이 아니라 (define 1+1 5)로 3개의 symbol이 존재하게 됩니다. (9/11 14:46 추가)
    - 프린트가 완료된 list의 Memory array는 free-list에 반환.
      * Hash table은 반환할 필요 없음
    - Lambda 식 변환은 구현하지 않으셔도 됩니다.