**REPORT**

**[Lab4]**



**과 목 : 프로그래밍언어론01**

**담당교수 : 송수환 교수님**

**학 과 : 컴퓨터공학과**

**학 번 : 2021111971**

**이 름 : 이재혁**

텍스트, 클립아트이(가) 표시된 사진

자동 생성된 설명

**To Do - Parser**

**Decl decl() -> <type> id [n]; 구현**

|  |  |
| --- | --- |
|  | 대괄호 안에 있는 n을 int로 계산해  size로 Decl객체(AST)를 생성합니다.  아이디, 타입, 크기  정보를 포함합니다.  크기 정보가 전달되어 배열의 선언임을 알 수 있습니다. |

**Stmt assignment() -> id[<expr1>] = <expr2>; 구현**

|  |  |
| --- | --- |
|  | id 배열의 <expr1>번째 index에  <expr2>의 값을 할당하는 Stmt입니다.  id의 index번째를 표현하는 Array객체(AST)를 생성합니다. |

**Expr factor() -> id'['<expr>']' 구현**

**텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

id 배열의 <expr>번째 index의 값을 표현하는 factor 입니다.

'[' 다음에 오는 Expr은 접근할 배열의 index의 표현식이기 때문에 Expr객체로 저장해 Array객체로 묶습니다.

**To Do - Sint**

**State Eval(Assginment a, State state) -> 배열 원소 할당 구현**

텍스트, 스크린샷, 소프트웨어, 폰트이(가) 표시된 사진

자동 생성된 설명

매개변수 a객체에 배열의 id 할당할 index의 위치(Expr)가 저장되어 있습니다.

state에서 아이디를 찾아 Value 배열 레퍼런스로 지정합니다.

Expr을 int값으로 계산해 해당 위치에 v를 할당합니다.

**State allocate(Decls ds, State state) -> 배열선언(state에 push) 구현**

텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

decl에 저장된 arraysize 크기의 Value 배열 레퍼런스를 생성합니다.

레퍼런스를 Value의 형태로 decl에 저장된 id와 함께 state에 push 합니다.

**Value V(Expr e, State state) -> id[<expr>]의 값 반환 구현**

텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

Array객체는 배열 id, Expr객체로 index를 저장하고 있습니다.

state에서 id를 탐색합니다. Value 배열은 Value의 형태로 저장되어 있어 arrValue() 메소드로 배열 형태로 반환해 레퍼런스를 선언합니다.

expr을 V()함수로 계산해 Value 객체로 변환 후 intValue() 메소드로 int 변수 index에 저장합니다.

해당 index에 해당하는 Value 객체를 반환합니다.

**결과분석**

**q1.s**

|  |  |
| --- | --- |
|  | int a[3];  state에 a와, Value[3]을 멤버 변수로 가지는 Value객체가 Pair를 이루어 state에 저장합니다.  a[1]=3; a[2]=7; a[3]=2;  a와 Expr객체(index)를 멤버변수로 가지는 Array객체를 생성합니다.  Array객체에 있는 a를 state에서 탐색하고  Value 배열을 찾습니다.  Expr을 계산해 index를 얻고 배열의 index 위치에 할당 표현식을 Value로 계산해 저장합니다.  print a[1];  a[1]에 해당하는 Array객체를 생성합니다.  state에서 a를 찾아 1번 위치의 Value를 print합니다. |

**q2.s**

**텍스트, 폰트, 스크린샷, 화이트이(가) 표시된 사진

자동 생성된 설명**

let의 decls에서 state에 a, Value(Value[5])가 저장됩니다.

while문을 반복하며 Identifier = a, Expr = i(반복하면서 변경됨)인 Array객체를 생성하고 state에서 a에 해당하는 Value[] 레퍼런스를 찾아 i의 위치에 i를 할당합니다. (a = [0, 1, 2, 3, 4])

while문을 반복하며 state에서 a에 해당하는 Value[] 레퍼런스를 찾아 i위치의 Value객체와 sum을 더해 sum에 할당합니다. (sum = sum + i | i : 0~4)

"Sum"과 sum에 저장된 값을 출력합니다.

**q3.s**

텍스트, 폰트, 화이트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

let의 decls에서 state에 a, Value(Value[5])가 저장됩니다.

Array 객체를 생성해 값 state에서 Value[]를 찾아 Value를 할당합니다.

(a = [1, 4, 3, 6, 2])

while문을 반복하며 a와 i로 Array객체를 생성하고 state에서 Value[]를 찾아 i번째 값을 max와 비교해 큰 값을 max에 저장합니다.

(max는 1, 3, 6)으로 변화합니다.

"Max"와 max에 저장된 값을 출력합니다.

**q4.s**

텍스트, 폰트, 화이트, 타이포그래피이(가) 표시된 사진

자동 생성된 설명

let의 decls에서 state에 abc, Value(Value[8])가 저장됩니다.

while문을 반복하며 abc와 i로 Array객체를 생성합니다.

i\*i - (i+3)\*3 - 3; 의 값을 계산해,

state에서 abc에 해당하는 Value[]를 찾아 i번째에 저장합니다.

(abc = [-12, -14, -14, -12, -8, -2, 6, 16])

while문을 반복하며 abc와 i(i : 0~7)로 Array객체를 생성합니다.

state에서 abc를 찾아 i번째의 값이 0보다 크다면 sum에 더해 저장합니다.

"Sum of positivev numbers"와 sum의 값을 출력합니다.

**q5.s**

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

string배열 ar1[6], int배열 ar2[6], bool배열 ar3[6]이 state에 저장됩니다.

Decl객체에 타입정보를 포함하고 있습니다.

ar1 = ["F", "A", "G", "U", "B", "E"]의 값으로 설정합니다.

while문으로

ar2 = [1, 2, 3, 4, 5, 6]

ar3 = [true, true, true, true, true, true] 의 값으로 설정합니다.

ar3[ar2[ar2[4]-4]] = ar3[2] = false;

ar2[1] = ar2[0] + ar2[1] = 1 + 2 = 3

ar2[2] = ar2[4] - ar2[ar2[ar2[1]]] = 5 - 5 = 0

ar2[3] = ar2[4]/(ar2[4] - ar2[1]) = 5/5-3 = 2

ar2[4] = ar2[1]\*ar2[3] - (ar2[0]\*2) = 3\*2 - 1\*2 = 4

ar2 = [1, 3, 0, 2, 4, 6]

while문으로

ar1[1], ar1[3], ar1[0], ar1[2], ar1[1](ar2[i] == 4, j = 4), ar1[4]를 출력합니다.

A U F G A B

ar1[j+1] = ar1[5]를 출력합니다.

E

ar3[3]을 출력합니다.

true