

Proposta de Resolução do Exame de ICL, 2010/2011, Época Normal

Diogo Sousa

16 de Janeiro de 2011

1 *Disclaimer*

Se algo estiver errado avisem, mas se houver erros que afectem a vossa nota no exame eu “*tou nem aí*”.

2 Parte 1

2.1 Alínea a)

- `ASTNum(int num)`
- `ASTAdd(ASTNode left, ASTNode right)`
- `ASTId(String id)`
- `ASTCall(ASTNode obj, String id, ASTNode arg)`
- `ASTAssign(String id, ASTNode exp)`
- `ASTSelf()`
- `ASTDecl(String id, ASTNode exp, ASTNode body)`
- `ASTObject(Map<String,ASTNode> fields, Map<String,Pair<String,ASTNode>> methods)`

2.2 Alínea b)

- `NumValue(int num)`
- `ObjValue(Map<String,Pair<String,ASTNode>> methods, Env env)`

2.3 Alínea c)

```
eval(ASTNum node, Env env)
    return new NumValue(node.num)

eval(ASTAdd node, Env env)
X    int l=((NumValue)eval(node.l,env)).num
```

```

X      int r=((NumValue)eval(node.r,env)).num
      return new NumValue(1+r)

      eval(ASTId node, Env env)
X      return env.find(node.id)

      eval(ASTSelf node, Env env)
X      return env.find("self")

      eval(ASTDecl node, Env env)
      Env newEnv=env.beginScope()
      newEnv.assoc(node.id,eval(node.exp,env))
      return eval(node.body,newEnv)

      eval(ASTAssign node, Env env)
X      env.update(node.id,eval(node.exp,env))

      eval(ASTObject node, Env env)
      ObjValue obj=new ObjValue()
      Env newEnv=env.beginScope()

      for (id,(argid,node')) in node.methods
        obj.addMethod(id,argid,node')

      for (id,node) in node.fields
        newEnv.assoc(id,eval(node,env))

      newEnv.assoc("self",obj)

      obj.setEnv(newEnv)

      return obj

      eval(ASTCall node, Env env)
X      ObjValue obj=eval(node.obj,env)
      Env newEnv=obj.env.beginScope()

      newEnv.assoc(obj.getMethodArgId(node.id),eval(node.arg,env))

      return eval(obj.getMethodBody(node.id),newEnv)

```

2.4 Alínea d)

Ver as cruzinhas (obviamente é preciso justificar)

3 Parte 4

3.1 Alínea a)

- Num
- $Obj(id_1(id'_1) : \tau_1 \rightarrow \tau'_1, \dots, id_n(id'_n) : \tau_n \rightarrow \tau'_n)$

3.2 Alínea b)

Identificador

$$\frac{id : \tau \in \Gamma}{\Gamma \vdash id : \tau} \quad (1)$$

Número

$$\Gamma \vdash num : Num \quad (2)$$

Soma

$$\frac{\Gamma \vdash e_1 : Num \quad \Gamma \vdash e_2 : Num}{\Gamma \vdash e_1 + e_2 : Num} \quad (3)$$

Declaração

$$\frac{\Gamma \vdash e : \gamma \quad \Gamma, id : \gamma \vdash b : \tau}{\Gamma \vdash \mathbf{decl} id = e \mathbf{in} b : \tau} \quad (4)$$

Afectação

$$\frac{\Gamma, id : \tau \vdash e : \tau}{\Gamma, id : \tau \vdash id : = e : \tau} \quad (5)$$

Objecto

$$\frac{\forall_i \Gamma \vdash e_i : \delta_i \quad \forall_j \Gamma, id''_j : \gamma_j, self : Obj(id'_1(id''_1) : \gamma_1 \rightarrow \tau_1, \dots) \vdash e'_j : \tau_j}{\Gamma \vdash [id_1 = e_1, \dots, id'_1(id''_1) : \gamma_1] : Obj(id'_1(id''_1) : \gamma_1 \rightarrow \tau_1, \dots)} \quad (6)$$

Self

$$\frac{self : Obj(id_1(id'_1) : \tau_1 \rightarrow \tau'_1, \dots, id_n(id'_n) : \tau_n \rightarrow \tau'_n) \in \Gamma}{\Gamma \vdash \mathbf{self} : Obj(id_1(id'_1) : \tau_1 \rightarrow \tau'_1, \dots, id_n(id'_n) : \tau_n \rightarrow \tau'_n)} \quad (7)$$

Chamada

$$\frac{\Gamma \vdash e : \gamma \quad id : Obj(\dots, id'(id'') : \gamma \rightarrow \tau, \dots) \in \Gamma}{\Gamma \vdash \mathbf{id.id'}(e) : \tau} \quad (8)$$