

Eksamen 2010

Det følgende er en del av grammatikken for et språk med funksjoner.

```
func → type func id signature stmt-list
type → int
type → bool
stmt-list → stmt-list stmt
stmt-list → stmt
stmt → assign-stmt
stmt → if-stmt
stmt → return-stmt
return-stmt → return exp
exp → id
exp → id + id
exp → true
exp → false
```

Ord i *kursiv* er ikke-terminaler, ord og tegn i **fet** skrift er terminal-symboler. *id* representerer et navn.

En funksjon har en type som enten er Integer eller Boolean. Reglen i dette språket er (ikke overraskende) at return-setningen (**return** *exp*) skal returnere en verdi med samme type som typen til funksjonen.

Fyll ut de tomme felter (markert med *) i attributtgrammatikken for de relevante deler av grammatikken på side 7 slik at attributtet *ok* for *return-stmt* er **true** hvis typen til returuttryket *exp* er det samme som typen til funksjonen, ellers **false**.

Du kan anta at `lookup-kind(id.name)` leverer den *type*, som er lagt inn i symboltabellen ved deklarasjonen av variabelen med navnet *id.name*.

Grammar Rule	Semantic Rule	
$func \rightarrow type \mathbf{func} id signature$ $stmt-list$		*
$type \rightarrow \mathbf{int}$	$type.type = Integer$	
$type \rightarrow \mathbf{bool}$	$type.type = Boolean$	
$stmt-list_1 \rightarrow stmt-list_2 stmt$		*
$stmt-list \rightarrow stmt$		*
$stmt \rightarrow return-stmt$		*
$return-stmt \rightarrow \mathbf{return} exp$	$return-stmt.ok =$	
$exp \rightarrow id$	$exp.type = lookup(id.name)$	
$exp \rightarrow id_1 + id_2$		*
$exp \rightarrow \mathbf{true}$	$exp.type = Boolean$	
$exp \rightarrow \mathbf{false}$	$exp.type = Boolean$	