

Manuel Utilisateur

Groupe gl27

1 Utilisation du compilateur et de l'extension

Afin de faire fonctionner le compilateur comme vous le voulez, il faut tout d'abord avoir un fichier d'extension `.deca`. Vous aurez sûrement besoin d'utiliser des options dans la commande `decac`, les voici :

- `-b` : Affiche la bannière de l'équipe.
- `-p` : Affiche la décompilation de l'arbre après sa construction
- `-v` : Arrête le programme après les vérifications syntaxique et contextuelle. À noter que l'on ne peut pas utiliser les options `-p` et `-v` en même temps.
- `-n` : Supprime les tests à l'exécution.
- `-r X` : Limite les registres utilisés pour ne se servir que des `X` premiers, avec $4 \leq X \leq 16$.
- `-d` : Lance le debug, en répétant l'option on obtient plus de traces de debug.
- `-P` : Permet la compilation parallèle s'il y a plusieurs fichiers sources.

En sortie du compilateur, on se retrouve avec un fichier de type `sortie.ass`. Afin de s'en servir, on doit l'exécuter avec la commande `ima`, qui permet de lancer les programmes en assembleur comme celui-ci.

Pour pouvoir faire fonctionner les programmes sous la forme d'un programme GBA, il suffit d'utiliser l'option `--gba` de la commande `decac`.

Il y a deux spécificités au niveau de la programmation :

- L'option `-r X` est un peu différente, en effet la Game Boy Advance ne peut aller que jusqu'à `X = 11`.
- En sortie, on se retrouve avec un fichier de type `sortie.s`, toutes les explications à ce sujet sont dans la documentation de l'extension.

2 Les messages d'erreur possibles

Les erreurs que vous pouvez obtenir avec les fichiers dans le dossier `/src/test/deca/syntax/invalid` sont :

- `LocationException` : Apparaît quand un token a mal été instancié, cette erreur est plutôt rare pour l'utilisateur.
- `DecacFatalError` : Cette erreur est présente quand il y a un problème avec la commande `decac`.

- **StackOverflowError** : On obtient ceci quand on essaie de rentrer une valeur trop élevée dans un registre.
- **AssertionError** : Si vous obtenez cette erreur, pensez à vérifier les conditions de vos **assert**.
- Dans le cas d'entiers trop grand, on a : **The integer (ou float) read is too large for the machine**
- **Exception** : Cette exception arrive quand l'erreur n'appartient pas aux catégories précédentes, pour trouver d'où vient l'erreur regardez le message d'erreur, voici quelques exemples :
 - mismatched input '...' expecting {...}
 - extraneous input '...' expecting '...'
 - missing '...' at '...'
 - no viable alternative at input "
 - left-hand side of assignment is not an lvalue
 - token recognition error at : "

Le message d'erreur que vous allez obtenir le plus souvent est l'erreur de type **ContextualError**, voici un tableau des différentes erreurs (que l'on peut retrouver dans le dossier `/src/test/deca/context/invalid`) :

Fichier	Message d'erreur
test12.assign.deca	AbstractExpr : RValue's type is not compatible with the expected type (type : " ... " ; expectedType : "...")
test10.ifThenElse.deca	AbstractExpr : Condition's type must be boolean (current : "...")
test3.op.arith.deca	AbstractOpArith : The types are incompatible for a Arithmetic Operation
test6.op.bool.deca	AbstractOpBool : The types are incompatible for a Boolean Operation
test7.op.comp.deca	AbstractOpCmp : Cannot compare "..." types
test.GEQ-class.deca	AbstractOpCmp : The type are incompatible for a "..." Operation
test1.bool.print.deca	AbstractPrint : Cannot print anything that is not an int, a float or a string
test.cast.void.deca	Cast : Cannot cast a void type
test.class.cast.incompatible.deca	Cast : The types are incompatible for a cast
test.class.definition.deca	DeclClass : The class "..." is already defined
test.inexistent.superclass.deca	DeclClass : The class "..." doesn't have a superclass or no exist
test.invalid.extends.deca	DeclClass : "..." 's superclass is not of type class
test.inexistent.type.deca	DeclClass : The type "..." doesn't exist
test.ident.attribute.deca	DeclField : The identifier "..." already exists and it is not an attribute
test.incorrect.field.deca	DeclField : Cannot declare a champ of type void
test.field.definition.deca	DeclField : Field "..." was already defined
test.override.invalid.signatures.deca	DeclMethod : The Override is impossible because signatures are not equals
test.invalid.override.deca	DeclMethod : The Override is impossible because the return type is not a subtype of that of the inherited method
test.ident.method.deca	DeclMethod : The identifier "..." already exists and it is not an method
test.invalid.method.definition.deca	DeclMethod : Method "..." was already defined
test.void.param.deca	DeclParam : Cannot declare a parameter of type void
test16.void.deca	DeclVar : Cannot declare a variable of type void
test17.multiple.decl.deca	DeclVar : Variable "..." was already defined
test.inexistent.var.deca	Identifier : Expression identifier is not declared in this scop
test.class.inexistent.deca	Identifier : Type "..." is not declared
test.instanceof.invalid.rightoperand.deca	InstanceOf : The right operand is not of type class
test.instanceof.invalid.leftoperand.deca	InstanceOf : The left operand is not of type class
test.duplicated.param.deca	ListDeclParam : Parameter "..." has been duplicated
test.incorrect.methodCall.deca	MethodCall : The object on which the method is called is not a class
test.environment.deca	MethodCall : The method "..." is not defined for the class
test.bad.parameters2.deca	MethodCall : Too many parameters have been given. Expected : "..." given : "..."
test.bad.parameters.deca	MethodCall : Too few parameters have been given. Expected : "..." given : "..."
test4.op.modulo.deca	Modulo : Modulo operator must be applied to two expression of int type (right expression's type : "..." left expression's type : "...")
test.invalid.new.deca	New : Cannot instantiate anything other than a class
test8.not.deca	Not : Not operator must be applied on a boolean expression (current type : "...")
test.selection.left.operand.deca	Selection : The left expression of the selection is not of type class
test.class.inexistent.field.deca	Selection : The field is not defined in class "..."
test.selection.right.operand.deca	Selection : The right identifier of the selection is not a field
test.visibility.cond1.deca	Selection : The field "..." is not visible
test.visibility.cond2.deca	Selection : The type of the expression ("...") is not a subtype of the current class "..."
test.this.main.deca	This : Cannot use this in main
test5.unary.minus.deca	UnaryMinus : Unary minus must be applied to an expression of type int or float (current type : "...")

3 Les limitations du compilateur

Le compilateur fonctionne principalement comme ce qui est spécifié, il n'y a juste pas l'option `-w` qui affiche les messages d'avertissement lors de la compilation.

Cependant, on ne peut pas initialiser un attribut dans une classe à l'aide d'une méthode de cette même classe.

```
class A {  
    int test(){return 0;}  
}  
class B {  
    A a = new A();  
    int x = a.test();  
}  
  
class A {  
    int test(){return 0;}  
    int x = this.test();  
}
```

Dans cet exemple, le code de gauche marche mais le code de droite ne va pas marcher. Aussi, il n'est pas possible d'accéder à des méthodes initialisées dans des classes déclarées après celle dans laquelle on se trouve.

4 Limitations de l'extension

Dans l'extension, certaines fonctionnalités du compilateur n'existent pas dans l'extension :

- L'extension ne peut pas encore gérer les nombres flottants, si cela arrive une erreur est renvoyée.
- L'extension marche uniquement dans le cadre d'une programmation sans objet, les objets ne sont pas encore implémentés.
- Les fonctions `readInt` et `readFloat` ne sont actuellement pas présentes.