ArnoldC

ArnoldC is an imperative programming language where the basic keywords are replaced with quotes from different Schwarzenegger movies. Parsing is done with [Parboiled](#) and [ASM](#) is used to generate the Java bytecode.

HelloWorld.arnoldc

```
IT'S SHOWTIME
TALK TO THE HAND "hello world"
YOU HAVE BEEN TERMINATED
```

Quick Start

```
java -jar ArnoldC.jar hello.arnoldc
java hello
```

Brief overview of the keywords

[False](#) I LIED
[True](#) NO PROBLEMO
[If](#) BECAUSE I'M GOING TO SAY PLEASE
[Else](#) BULLSHIT
[EndIf](#) YOU HAVE NO RESPECT FOR LOGIC
[While](#) STICK AROUND
[EndWhile](#) CHILL
PlusOperator GET UP
[MinusOperator](#) GET DOWN
[MultiplicationOperator](#) YOU'RE FIRED
[DivisionOperator](#) HE HAD TO SPLIT
[ModuloOperator](#) I LET HIM GO
[EqualTo](#) YOU ARE NOT YOU YOU ARE ME
[GreaterThan](#) LET OFF SOME STEAM BENNET
[Or](#) CONSIDER THAT A DIVORCE
[And](#) KNOCK KNOCK
[DeclareMethod](#) LISTEN TO ME VERY CAREFULLY
[NonVoidMethod](#) GIVE THESE PEOPLE AIR
[MethodArguments](#) I NEED YOUR CLOTHES YOUR BOOTS AND YOUR MOTORCYCLE

Return I'LL BE BACK
EndMethodDeclaration HASTA LA VISTA, BABY
CallMethod DO IT NOW
AssignVariableFromMethodCall GET YOUR ASS TO MARS
DeclareInt HEY CHRISTMAS TREE
SetInitialValue YOU SET US UP
BeginMain IT'S SHOWTIME
EndMain YOU HAVE BEEN TERMINATED
Print TALK TO THE HAND
ReadInteger I WANT TO ASK YOU A BUNCH OF QUESTIONS AND I WANT TO HAVE THEM ANSWERED
IMMEDIATELY
AssignVariable GET TO THE CHOPPER
SetValue HERE IS MY INVITATION
EndAssignVariable ENOUGH TALK

Main method

Every ArnoldC program must have a main method. The form is:

```
IT'S SHOWTIME
[statements]
YOU HAVE BEEN TERMINATED
```

The simplest possible ArnoldC program that does nothing

```
IT'S SHOWTIME
YOU HAVE BEEN TERMINATED
```

Statements

Printing

The statement `TALK TO THE HAND` is used to print strings or variables.

Printing a string
```
TALK TO THE HAND "happy families are all alike"
```

Printing a variable
```
TALK TO THE HAND myvar
```

Declaring variables

The only variable type in ArnoldC is a 16bit signed integer. A value must be given to the variable when it is declared.

```
HEY CHRISTMAS TREE variablename
YOU SET US UP initialvalue
```

The "macros" `@I LIED` and `@NO PROBLEMO` correspond to the values 0 and 1
Initializing two variables with the values of 0 and 1

```
HEY CHRISTMAS TREE varFalse
YOU SET US UP @I LIED
HEY CHRISTMAS TREE varTrue
YOU SET US UP @NO PROBLEMO
```

Assigning variables

Variable assignment is done using the pattern

```
GET TO THE CHOPPER myvar
HERE IS MY INVITATION firstOperand
[operations]
ENOUGH TALK
```

The `HERE IS MY INVITATION` sets a value on the top of the stack. The rest of the operations always apply to the current value of the stack which is finally assigned to the `myvar` variable.

Arithmetic operations

(All the arithmetic operations have the same precedence)

Plus
```
GET UP operand
```
Minus
```
GET DOWN operand
```
Multiplication
```
YOU'RE FIRED operand
```
Division
```
HE HAD TO SPLIT operand
```

Example a = (4 + b) * 2
```
GET TO THE CHOPPER a
HERE IS MY INVITATION 4
GET UP b
YOU'RE FIRED 2
ENOUGH TALK
```

Logical operations

True statements result the value of 1 and false statements the value of 0.

Equal to
```
YOU ARE NOT YOU YOU ARE ME operand
```

Greater than
```
LET OFF SOME STEAM BENNET operand
```

Or
```
CONSIDER THAT A DIVORCE operand
```

And
```
KNOCK KNOCK operand
```

Example a = (b + 5) > c
```
GET TO THE CHOPPER a
HERE IS MY INVITATION b
GET UP 5
LET OFF SOME STEAM BENNET c
ENOUGH TALK
```

Example a = (b || c) && d
```
GET TO THE CHOPPER a
HERE IS MY INVITATION b
CONSIDER THAT A DIVORCE c
KNOCK KNOCK d
ENOUGH TALK
```

Condition statements

The condition branch is executed if the value is anything other than 0.

If

```
if (value) [statements]
BECAUSE I'M GOING TO SAY PLEASE value
[statements]
YOU HAVE NO RESPECT FOR LOGIC
```

If Else

```
if (value) [statements] else [statements]
BECAUSE I'M GOING TO SAY PLEASE value
[statements]
BULLSHIT
[statements]
YOU HAVE NO RESPECT FOR LOGIC
```

Example

```
if(a) print "a is true"
else print "a is not true"
BECAUSE I'M GOING TO SAY PLEASE a
TALK TO THE HAND "a is true"
BULLSHIT
TALK TO THE HAND "a is not true"
YOU HAVE NO RESPECT FOR LOGIC
```

Example The value of the condition statement must be calculated beforehand `if(a > b)`
`print "a is greater than b"`

```
GET TO THE CHOPPER result
HERE IS MY INVITATION a
LET OFF SOME STEAM BENNET b
ENOUGH TALK
BECAUSE I'M GOING TO SAY PLEASE result
TALK TO THE HAND "a is greater b"
YOU HAVE NO RESPECT FOR LOGIC
```

While

```
STICK AROUND value
[statements]
CHILL
```

Example A full program printing numbers 1-10

```
IT'S SHOWTIME
HEY CHRISTMAS TREE isLessThan10
YOU SET US UP @NO PROBLEMO
HEY CHRISTMAS TREE n
YOU SET US UP 0
STICK AROUND isLessThan10
GET TO THE CHOPPER n
HERE IS MY INVITATION n
GET UP 1
ENOUGH TALK
TALK TO THE HAND n
GET TO THE CHOPPER isLessThan10
HERE IS MY INVITATION 10
LET OFF SOME STEAM BENNET n
ENOUGH TALK
CHILL
YOU HAVE BEEN TERMINATED
```

Methods

Methods must be defined outside the main method. The method arguments are defined with I NEED YOUR CLOTHES YOUR BOOTS AND YOUR MOTORCYCLE arg. The one-liner must be added for each argument separately. GIVE THESE PEOPLE AIR at the end of the "method declaration" indicates the method is non-void. I'LL BE BACK is the keyword for RETURN and it can take an operand depending whether the method is void or not.
A void method

```
LISTEN TO ME VERY CAREFULLY methodName
[Statements]
HASTA LA VISTA, BABY
```

A non-void method that takes two parameters

```
LISTEN TO ME VERY CAREFULLY methodName
I NEED YOUR CLOTHES YOUR BOOTS AND YOUR MOTORCYCLE arg1
I NEED YOUR CLOTHES YOUR BOOTS AND YOUR MOTORCYCLE arg2
GIVE THESE PEOPLE AIR
[Statements]
HASTA LA VISTA, BABY
```

Running Programs:

Try it yourself

hello.arnoldc

```
IT'S SHOWTIME
TALK TO THE HAND "hello world"
YOU HAVE BEEN TERMINATED
```

```
java -jar ArnoldC.jar hello.arnoldc
java hello
```