

Προγραμματιστική άσκηση:

Η γλώσσα προγραμματισμού Ciscal

Η Ciscal είναι μια μικρή γλώσσα προγραμματισμού. Παρόλο που οι προγραμματιστικές της ικανότητες είναι μικρές, η εκπαιδευτική αυτή γλώσσα περιέχει πλούσια στοιχεία και η κατασκευή του μεταγλωττιστή της έχει να παρουσιάσει αρκετό ενδιαφέρον, αφού περιέχονται σε αυτήν πολλές εντολές που χρησιμοποιούνται από άλλες γλώσσες, καθώς και κάποιες πρωτότυπες. Η Ciscal περιέχει συναρτήσεις και διαδικασίες, μετάδοση παραμέτρων με αναφορά και τιμή, αναδρομικές κλήσεις, κλπ. Επίσης, επιτρέπει φώλιασμα στη δήλωση συναρτήσεων και διαδικασιών κάτι που λίγες γλώσσες υποστηρίζουν (το υποστηρίζει η Pascal, δεν το υποστηρίζει η C).

Από την άλλη όμως πλευρά, η Ciscal δεν υποστηρίζει βασικά προγραμματιστικά εργαλεία όπως η δομή for καθώς και πραγματικούς αριθμούς και συμβολοσειρές. Οι παραλήψεις αυτές έχουν γίνει ώστε να απλουστευτεί η διαδικασία κατασκευής του μεταγλωττιστή, μία απλούστευση όμως που έχει να κάνει μόνο με τη μείωση των γραμμών του κώδικα και όχι με τη δυσκολία κατασκευής του, αφού οι δομές που υποστηρίζει είναι αυτές που παρουσιάζουν περισσότερο ενδιαφέρον όσον αφορά την υλοποίηση, ενώ η διάκριση ανάμεσα σε πραγματικούς και ακραίους αριθμούς δεν έχει να προσφέρει κάτι εκπαιδευτικά.

Λεκτικές μονάδες

Το αλφάβητο της Ciscal αποτελείται από τα μικρά και κεφαλαία γράμματα της λατινικής αλφαβήτου («A»,...,«Z» και «a»,...,«z»), τα αριθμητικά ψηφία («0»,...,«9»), τα σύμβολα των αριθμητικών πράξεων («+», «-», «*», «/»), τους τελεστές συσχέτισης («<», «>», «=», «<=», «>=», «<>»), το σύμβολο ανάθεσης «:=», τους διαχωριστές («;», «,») καθώς και τα σύμβολα ομαδοποίησης («{»,«}»,«(»,«)»,«[»,«]») και διαχωρισμού σχολίων («\»,«*»,«*\»). Μερικές λέξεις είναι δεσμευμένες:

and	declare	do	else	enddeclare
exit	procedure	function	print	call
if	in	inout	not	select
program	or	return	while	default

Οι λέξεις αυτές δε μπορούν να χρησιμοποιηθούν ως μεταβλητές. Οι σταθερές της γλώσσας είναι ακέραιες σταθερές που αποτελούνται από προαιρετικό πρόσημο και από μία ακολουθία αριθμητικών ψηφίων.

Τα αναγνωριστικά της γλώσσας είναι συμβολοσειρές που αποτελούνται από γράμματα και ψηφία, αρχίζοντας όμως από γράμμα. Ο μεταγλωττιστής λαμβάνει υπόψη του μόνο τα τριάντα πρώτα γράμματα. Οι λευκοί χαρακτήρες (tab, space, return) αγνοούνται και μπορούν να χρησιμοποιηθούν με οποιονδήποτε τρόπο χωρίς να επηρεάζεται η λειτουργία του μεταγλωττιστή, αρκεί βέβαια να μη βρίσκονται μέσα σε δεσμευμένες λέξεις, αναγνωριστικά, σταθερές, ή άλλες δεσμευμένες λέξεις. Το ίδιο ισχύει και για τα σχόλια, τα οποία πρέπει να βρίσκονται ανάμεσα στα σύμβολα * και *\.

Τύποι και δηλώσεις μεταβλητών

Ο μοναδικός τύπος δεδομένων που υποστηρίζει η Ciscal είναι οι ακέραιοι αριθμοί. Οι ακέραιοι αριθμοί πρέπει να έχουν τιμές από -32768 έως 32767. Η δήλωση γίνεται με την εντολή declare. Ακολουθούν τα ονόματα των αναγνωριστικών χωρίς καμία άλλη δήλωση, αφού γνωρίζουμε ότι πρόκειται για ακέραιες μεταβλητές. Οι μεταβλητές χωρίζονται μεταξύ τους με κόμματα. Το τέλος της δήλωσης των μεταβλητών γίνεται με την εντολή enddeclare.

Τελεστές και εκφράσεις

Η προτεραιότητα των τελεστών από τη μεγαλύτερη στη μικρότερη είναι:

- (1) Μοναδικοί «not»
- (2) Πολλαπλασιαστικοί «*», «/»
- (3) Μοναδικοί προσθετικοί «+», «-»
- (4) Δυαδικοί προσθετικοί «+», «-»
- (5) Σχεσιακοί «=», «<», «>», «<>», «<=», «>=»
- (6) Λογικοί «and», «or»

Εντολές

- Εκχώρηση

Id := expression

Χρησιμοποιείται για την ανάθεση της τιμής μίας μεταβλητής ή μίας σταθεράς, ή μίας έκφρασης σε μία μεταβλητή.

- Απόφασης

```
if (condition)
{...}
[ else
{...} ]
```

Η εντολή απόφασης εκτιμάει εάν ισχύει η συνθήκη condition και εάν πράγματι ισχύει εκτελούνται οι εντολές που ακολουθούν. Το else δεν αποτελεί υποχρεωτικό τμήμα της εντολής και γι' αυτό βρίσκεται σε αγκύλη. Οι εντολές που το ακολουθούν εκτελούνται εάν η συνθήκη condition δεν ισχύει. Τα άγκιστρα δεν είναι υποχρεωτικά, όταν έχουμε μόνο μία εντολή.

- Επανάληψης

```
do
{...}
while (condition)
```

Η εντολή επανάληψης εκτελεί τις εντολές που βρίσκονται ανάμεσα στα άγκιστρα και τις επαναλαμβάνει έως ότου η συνθήκη condition δεν ισχύει ή εκτελεστεί η εντολή exit. Με την εντολή exit η εκτέλεση μεταφέρεται έξω από το βρόχο.

```
while (condition)
{...}
```

Η εντολή επανάληψης εκτελεί τις εντολές που βρίσκονται ανάμεσα στα άγκιστρα και τις επαναλαμβάνει έως ότου η συνθήκη condition δεν ισχύει.

- Επιλογής

```
select (id)
1: {...}
2: {...}
3: {...}
default: {...}
```

Ανάλογα με την τιμή του id εκτελείται μία από τις επιλογές. Αν id=1 η πρώτη, αν id=2 η δεύτερη, κ.ο.κ.. Αν καμία από τις επιλογές δεν ισούται με την τιμή του id, τότε εκτελείται η default. Προσοχή οι αριθμοί που εκφράζουν τις επιλογές πρέπει να ξεκινούν από το 1 και να είναι συνεχόμενοι.

- Επιστροφής

```
return (expression)
```

Χρησιμοποιείται μέσα σε συναρτήσεις για να επιστραφεί το αποτέλεσμα της συνάρτησης.

- Κλήσης Συναρτησης

```
call function_name(actual_parameters)
```

Χρησιμοποιείται μέσα σε συναρτήσεις για να επιστραφεί το αποτέλεσμα της συνάρτησης.

- Εξόδου

```
print(expression)
```

Εμφανίζει στην οθόνη το αποτέλεσμα της αποτίμησης του expression

Υποπρογράμματα

Η Ciscal υποστηρίζει συναρτήσεις και διαδικασίες

```
function id (formal_parameters)
{
    variable declarations
    subprograms
    sequence of statements
}

procedure id (formal_parameters)
{
    variable declarations
    subprograms
    sequence of statements
}
```

Η «formal_parameters» είναι η λίστα των τυπικών παραμέτρων. Οι διαδικασίες και οι συναρτήσεις μπορούν να φωλιάσουν η μία μέσα στην άλλη και οι κανόνες εμβέλειας είναι όπως της PASCAL. Η επιστροφή της τιμής μιας συνάρτησης γίνεται με την εντολή return. Η κλήση μιας διαδικασίας γίνεται με την εντολή call.

Μετάδοση παραμέτρων

Η Ciscal υποστηρίζει δύο τρόπους μετάδοσης παραμέτρων:

- (1) με σταθερή τιμή. Δηλώνεται με τη λεκτική μονάδα `in`. Αλλαγές στην τιμή της δεν επιστρέφονται στον καλόν πρόγραμμα
- (2) με αναφορά. Δηλώνεται με τη λεκτική μονάδα `inout`. Κάθε αλλαγή στη τιμή της μεταφέρεται και στο πρόγραμμα που κάλεσε τη συνάρτηση.

Μορφή προγράμματος

```
program id
{
    variable declarations
    subprograms
    sequence of statements
}
```