

Αρχές Γλωσσών Προγραμματισμού και Μεταφραστών



Απαντήσεις Project

Αθανασία Ζεκυριά (1059660) — Λουδάρος Ιωάννης (1067400)
Τσίκελης Ιωάννης (1067407) — Χριστίνα Κρατημένου (1067495)

Γενικές Πληροφορίες

Στις επόμενες σελίδες αναλύονται οι απαντήσεις τις ομάδας μας στο Project του μαθήματος “Αρχές Γλωσσών Προγραμματισμού και Μεταφραστών”. Σε αυτή την σελίδα έχετε πρόσβαση σε γενικές πληροφορίες γύρω από το Project αλλά και γύρω από τα μέλη της ομάδας μας.



Για να έχετε πρόσβαση στην τελευταία έκδοση των απαντήσεων μπορείτε να σκανάρετε το παραπάνω QR Code ή να χρησιμοποιήσετε το παρακάτω κουμπί.

Πατήστε Εδώ

Για την υλοποίηση του Project εργαστήκαμε σε μια ομάδα 4 ατόμων. Κάναμε συναντήσεις σε εβδομαδιαία βάση για να αξιολογήσουμε την πρόοδο μας αλλά και να καθορίσουμε τους επόμενους μας στόχους. Όλοι είχαμε επίγνωση των εργασιών που πραγματοποιούν οι συνεργάτες μας και δίναμε feedback ο ένας στον άλλον με αποτέλεσμα να προχωράμε συνεκτικά. Η ομάδα αποτελείται από τα εξής άτομα:

Ιωάννης Λουδάρος (1067400)

Χριστίνα Κρατημένου (1067495)

Αθανασία Ζεκυριά (1059660)

Ιωάννης Τσικέλης (1067407)

Παρακάτω υπάρχουν αναλυτικότερες πληροφορίες για τα μέλη της ομάδας μας.



Ιωάννης Τσικέλης
1067407

Email

Άλλα στοιχεία και μια μικρή περιγραφή, ότι θέλετε βάζετε απλά πείτε μου και θα το βρούμε. Από κάτω έχει έναν σύνδεσμο για κάτι (εγώ θα βάλω LinkedIn)

More



Αθανασία Ζεκυριά
1059660

Email

Άλλα στοιχεία και μια μικρή περιγραφή, ότι θέλετε βάζετε απλά πείτε μου και θα το βρούμε. Από κάτω έχει έναν σύνδεσμο για κάτι (εγώ θα βάλω LinkedIn)

More



Ιωάννης Λουδάρος
1067400

Email

Άλλα στοιχεία και μια μικρή περιγραφή, ότι θέλετε βάζετε απλά πείτε μου και θα το βρούμε. Από κάτω έχει έναν σύνδεσμο για κάτι (εγώ θα βάλω LinkedIn)

More



Χριστίνα Κρατημένου
1067495

Email

Άλλα στοιχεία και μια μικρή περιγραφή, ότι θέλετε βάζετε απλά πείτε μου και θα το βρούμε. Από κάτω έχει έναν σύνδεσμο για κάτι (εγώ θα βάλω LinkedIn)

More

Περιεχόμενα

1. Συντακτικός ορισμός της Γλώσσας σε BNF	3
2. Λεξικός Αναλυτής	6
3. Συντακτικός Αναλυτής	
4.	
5.	

Συντακτικός ορισμός της Γλώσσας σε BNF

Θεμελιώδεις έννοιες

Τύποι μεταβλητών

```
<digit> ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
<integer> ::= [ - ] ( <digit> | <digit> <integer> )
<p-integer> ::= <digit> | <digit> <integer>
<char> ::= a | ... | z | A | ... | Z
<punctuation> ::= .
          | ?
          | !
          | ,
          | ;
          | :
          | -
          | "["
          | "]"
          | "{"
          | "}"
          | "
          | "
          | "("
          | ")"
          | '
```

```
<string> ::= <letter> <string>
          | <digit> <string>
          | <punctuation> <string>
          | <operator> <string>
          | <letter>
          | <digit>
```

```
<var-name> ::= ( ( _ | <letter> ) [ <var-name> ] )
          | <letter> <var-name>
          | <digit> <var-name>
          | _ <var-name>
          | <letter>
          | <digit>
```

```
<variable> ::= <type> <var-name> [ "[" <p-integer> "]" ]
```

```
<variables> ::= <variable> | <variable>, <variables>
```

Αν προλάβουμε,
Συντακτικά
Διαγράμματα

Οι μετβλητές
ακολουθούν την
ονοματολογία της C.

Operators

`<operator> ::= <a-operator> | <l-operator>`

`<a-operator> ::= +`

| -
| ^
| *
| /

`<l-operator> ::= AND`

| OR
| ==
| !=
| <
| >

Keywords and Special Characters

`<ws> ::= \t | \n | “ ”`

`<type> ::= INT | CHAR`

Δομικά block ενός προγράμματος

Δηλώσεις

`<program title> ::= PROGRAM <string> \n`

`<func-declaration> ::= FUNCTION <string> “(“ <variables> “)“`

`<var-declaration> ::= VARS <variables> ;`

Εντολές

`<a-expression> ::= <integer>`

| `<var-name>`
| `<a-expression> <a-operator> <a-expression>`
| `“(“ <a-expression> “)“`
| `- <a-expression>`

`<l-expression> ::= <integer>`

| `<var-name>`
| `<a-expression>`
| `<l-expression> <l-operator> <l-expression>`
| `“(“ <l-expression> “)“`

`<assignment> ::= (<var-name>) = (<a-expression> | <char>) ;`

`<print> ::= PRINT “(“ <string> [, “[“ <var-name> “]”] “)“ ;`

Δομές Επανάληψης

`<while> ::= WHILE “(“ <l-expression> “)“ \n`

`<statements> \n`

`ENDWHILE`

`<for> ::= FOR counter:= <integer> TO <integer> STEP <integer> \n`

```

<statements> \n
ENDFOR

Δομές Απόφασης
<if> ::= IF "(" <l-expression> ")" THEN \n
    <statements> \n
    { ELSEIF \n
        <statements> \n
    } [ ELSE <statements> \n ]
ENDIF

<switch> ::= SWITCH "(" <l-expression> ")" \n
CASE "(" <l-expression> ")" : \n
    <statements> \n
{ CASE "(" <l-expression> ")" : \n
<statements> \n }
[ DEFAULT: \n
<statements> \n ]
ENDSWITCH

```

Σχόλια

<comment> ::= % <string>

Πιο αφηρημένες έννοιες

```

<statements> ::= <assignment> [<statements>]
| <print> [<statements>]
| <while> [<statements>]
| <for> [<statements>]
| <if> [<statements>]
| <switch> [<statements>]
| <comment> [<statements>]
| BREAK;
| <string> "(" <variables> ")" \n
    "{" <statements> "}"
| "{" <statements> "}"
<function> ::= <func-declaration> \n
    <var-declaration> \n
    <statements> \n
    RETURN ( <a-expression> | <string> ) \n
    END_FUNCTION
<main> ::= STARTMAIN \n
    <var-declaration> \n
    <statements>
    ENDMAIN
<program> ::= <program title> { <function> } <main>

```

Αν προλάβουμε,
Συντακτικά
Διαγράμματα

Αυτή την γραμμή την χρησιμοποιούμε ώστε ο χρήστης να μπορεί να χρησιμοποιεί τυχόν συναρτήσεις που ορίζει.

Υποσημείωση : Σε κάποια σημεία είναι απαραίτητο, μέσα σε παραγωγές να χρησιμοποιήσουμε σύμβολα που κανονικά είναι δεσμευμένα για την σημειογραφία Backus-Naur. Σε αυτή την περίπτωση αντί για απλή αναγραφή του στοιχείου, περικλείεται σε κόκκινα εισαγωγικά (πχ ““)