**PLanG**

**a A PLanG környezet futtatása**

* A [plang.jar](https://drive.google.com/open?id=0B5ByL7MsfiL6c2hfQ2pNSEJiWmM) fájlt le lehet tölteni és haza lehet vinni, a java -jar plang paranccsal lehet elindítani
* A futtatáshoz szükség van egy Java 5 (vagy újabb) futtató környezetre. Szükség esetén az [Oracle oldaláról](http://java.com/en/download/) lehet letölteni.

**A programokban használható típusok és műveleteik**

**Logikai típus**

Típusértékek:

IGAZ, HAMIS

Műveletek:

· Értékadás: LogValt := LogKif

Logikai eredményű kifejezések:

· Logikai konstansok és logikai típusú változók

· (LogKif): zárójelezés

· NEM LogKif: logikai *nem*

· LogKif1 ÉS LogKif2: logikai *és*

· LogKif1 VAGY LogKif2: logikai *vagy*

**Egész szám típus**

Típusértékek:

Pozitív és negatív egész számok és a nulla (0, 1, 2, -1, -2, ...)

Műveletek:

· Értékadás: EgValt := EgKif

· Beolvasás: BE: EgValt

· Kiírás: KI: EgKif

Egész eredményű kifejezések:

· Egész konstansok és egész típusú változók

· (EgKif): zárójelezés

· - EgKif: negáció

· |EgKif|: abszolút érték

· EgKif1 + EgKif2: összeadás

· EgKif1 - EgKif2: kivonás

· EgKif1 \* EgKif2: szorzás

· EgKif1 DIV EgKif2: egész osztás (az eredmény törtrésze elveszik)

· EgKif1 MOD EgKif2: osztási maradék

· RND EgKif: véletlenszám előállítása a [0..EgKif) intervallumban

Valós eredményű kifejezés:

· VALÓS EgKif: ugyanaz az érték valós típussal

Logikai eredményű kifejezések:

· EgKif1 = EgKif2: egyenlőségvizsgálat

· EgKif1 /= EgKif2: nem egyenlő

· EgKif1 < EgKif2: kisebb

· EgKif1 > EgKif2: nagyobb

· EgKif1 <= EgKif2: kisebb vagy egyenlő

· EgKif1 >= EgKif2: nagyobb vagy egyenlő

**Valós szám típus**

Típusértékek:

Pozitív és negatív valós számok és a nulla (0, 3.141, -0.06, ...)

Műveletek:

· Értékadás: ValValt := ValKif

· Beolvasás: BE: ValValt

· Kiírás: KI: ValKif

Valós eredményű kifejezések:

· Lebegőpontos konstansok és valós típusú változók

· Egész típusú kifejezések

· (ValKif): zárójelezés

· - ValKif: negáció

· ValKif1 + ValKif2: összeadás

· ValKif1 - ValKif2: kivonás

· ValKif1 \* ValKif2: szorzás

· ValKif1 / ValKif2: osztás

· ValKif1 ^ ValKif2: hatványozás

· |ValKif|: abszolút érték

· SIN ValKif: szinusz; COS, TAN, ARCSIN, ARCCOS, ARCTAN, LOG, EXP: értelemszerűen

Egész eredményű kifejezések:

· EGÉSZ ValKif: törtrész levágása

· KEREK ValKif: kerekítés

Logikai eredményű kifejezések:

· ValKif1 = ValKif2: egyenlőségvizsgálat

· ValKif1 /= ValKif2: nem egyenlő

· ValKif1 < ValKif2: kisebb

· ValKif1 > ValKif2: nagyobb

· ValKif1 <= ValKif2: kisebb vagy egyenlő

· ValKif1 >= ValKif2: nagyobb vagy egyenlő

**Karakter típus**

Típusértékek:

Betűk, számjegyek, írásjelek, szóköz, sorvége jel ('A', 'x', ' ', '?', '@', SV)

Műveletek:

· Értékadás: KarValt := KarKif

· Beolvasás: BE: KarValt

· Kiírás: KI: KarKif

Karakter eredményű kifejezések:

· Karakter konstansok, karakter típusú változók

· (KarKif): zárójelezés

· NAGY KarKif: nagybetűvé alakít

· KIS KarKif: kisbetűvé alakít

Logikai eredményű kifejezések:

· KarKif1 = KarKif2: egyenlőségvizsgálat

· KarKif1 /= KarKif2: nem egyenlő

· KarKif1 < KarKif2: ábécésorrendben előtte

· KarKif1 > KarKif2: ábécésorrendben utána

· KarKif1 <= KarKif2: előtte vagy egyenlő

· KarKif1 >= KarKif2: utána vagy egyenlő

· BETŰ KarKif: betű-e?

· SZÁM KarKif: számjegy-e?

**Szöveg típus**

Típusértékek:

Karakterek (akár üres) sorozatai ("", "alma", "több szó")

Műveletek:

· Értékadás: SzValt := SzKif

· Karakter megváltoztatása: SzValt[EgKif] := KarKif

· Sor beolvasása: BE: SzValt

· Kiírás: KI: SzKif

Szöveg eredményű kifejezések:

· Szövegkonstansok, szöveg típusú változók

· (SzKif): zárójelezés

· SzKif1 + SzKif2: szövegek összefűzése

· KarKif + SzKif: karakter szöveg elé fűzése

· SzKif + KarKif: karakter szöveg után fűzése

· SzKif[EgKif1:EgKif2]: résszöveg lekérdezés, az EgKif1 pozíción kezdődő és EgKif2 pozíció előtt végződő rész az eredmény

Egész eredményű kifejezések:

· |SzKif|: hossz lekérdezése

· SzKif @ KarKif: karakter első előfordulásának helye (a karakter indexe; ha nem fordulelő, akkor |SzKif| az eredmény)

· SzKif1 @ SzKif2: résszöveg első előfordulásának helye (a résszöveg első karakterének az indexe; ha nem fordul elő, akkor |SzKif1| az eredmény)

Karakter eredményű kifejezések:

· SzKif[EgKif]: karakter lekérdezése

Logikai kifejezések:

· SzKif1 = SzKif2: egyenlőségvizsgálat

· SzKif1 /= SzKif2: nem egyenlő

· SzKif1 < SzKif2: ábécésorrendben előtte

· SzKif1 > SzKif2: ábécésorrendben utána

· SzKif1 <= SzKif2: előtte vagy ugyanaz

· SzKif1 >= SzKif2: utána vagy ugyanaz

**Tömb típus**

Jelölés:

*Alaptípus*[Méret1]...[Méretdim], ahol Méreti egész konstans.

Típusértékek:

· Ha dim=1: Méret1 darab *Alaptípus* típusú érték (Méret1 elemű vektor)

· Ha dim=2: Méret1 darab *Alaptípus*[Méret2] típusú tömb (Méret1 × Méret2-es mátrix)

· Ha dim=*n*: Méret1 darab *Alaptípus*[Méret2]...[Méret*n*] típusú tömb (*n* dimenziós tömb)

Műveletek:

· Értékadás: TombValt := TombKif

· Értékadás tömbelemnek: TombValt[EgKif] := Kif

· Értékadás tömbelemnek: TombValt[EgKif1]...[EgKif*n*] := Kif

· Kiírás: KI: TombKif

Egész eredményű kifejezések:

· |TombKif|: elemszám lekérdezése (Méret1 az eredmény)

Tömb eredményű kifejezések:

· TombVal[EgKif1]...[EgKif*n*]: tömbelem lekérdezés; ha *n*<dim, akkor az eredmény egy dim-*n* dimenziós tömb

*Alaptípus* eredményű kifejezések:

· TombVal[EgKif1]...[EgKif*dim*]: tömbelem lekérdezés

**Fájl típusok**

Jelölés:

BEFÁJL és KIFÁJL

Típusértékek:

· BEFÁJL: karaktersorozatot tartalmazó fájlok (névvel azonosítva), illetve lezárt fájl

· KIFÁJL: karakterekkel feltölthető fájlok (névvel azonosítva), illetve lezárt fájl

Műveletek:

· Megnyitás: MEGNYIT FileValt: "fájlnév.txt"

· Lezárás: LEZÁR FileValt

· Írás: KI KiFileValt: ...

· Olvasás: BE BeFileValt: ...

Logikai eredményű kifejezés:

· VÉGE BeFileValt: akkor ad vissza igazat, ha a legutóbbi beolvasó művelet nem sikerült (például mert nincs több adat a fájlban).

**Műveleti sorrend**

1. Prefix műveletek: NEM, -kif, SIN, ...
2. Hatványozás: ^, keresés: @
3. Multiplikatív műveletek: \*, /, DIV, MOD
4. Additív műveletek: +, -
5. Relációs műveletek: =, /=, <, ...
6. Logikai műveletek: ÉS, VAGY

**A programok felépítésének szabályai**

A szabályok leírásakor használt jelölések:

* Az idézőjelek közötti szövegek a programszöveg építőelemei, egy az egyben kerülnek bele a forráskódba
* A szögletes zárójelek elhagyható részeket jelölnek
* A kapcsos zárójelek ismétlést jelölnek: a zárójelek közötti teljes rész egymás után tetszőleges sokszor szerepelhet (akár nullaszor is, tehát el is hagyható).
* A Betű-k a magyar ABC kis- és nagybetűi, a Számjegy-ek a számjegyek 0-tól 9-ig, az EgyébJel-ek tetszőleges egyéb karakterek (szóköz, írásjelek, és bármi amit ki lehet csalni a billentyűzetből). A "”" a sima felső jobbra dőlő dupla idézőjelet, a "’" a sima felső jobbra dőlő aposztrófot jelöli.

**A szabályok**

|  |  |
| --- | --- |
| *Program* ::= | { *Megjegyzés* ... } "PROGRAM" Azonosító *Deklarációk* *Utasítások* "PROGRAM\_VÉGE" |
| *Deklarációk ::=* | [ "VÁLTOZÓK" ":" *VáltozóDeklaráció* { "," *VáltozóDeklaráció* ... } ] |
| *VáltozóDeklaráció ::=* | Azonosító { "," Azonosító ... } ":" *Típus* |
| *Utasítások ::=* | *Utasítás* { [ "," ] *Utasítás* ... } |
| *Utasítás ::=* | *Értékadás* | *Elágazás* | *Ciklus* | *InputOutput* | *Megjegyzés* |
| *Értékadás ::=* | *Balérték* ":=" *Kifejezés* |
| *Elágazás ::=* | "HA" *Kifejezés* "AKKOR" *Utasítások* [ "KÜLÖNBEN" *Utasítások* ] "HA\_VÉGE" |
| *Ciklus ::=* | "CIKLUS" *Utasítások* "AMÍG" *Kifejezés* | "CIKLUS AMÍG" *Kifejezés* *Utasítások* "CIKLUS\_VÉGE" |
| *InputOutput ::=* | "BE" [ Azonosító ] ":" *Balérték* { "," *Balérték* ... } | "KI" [ Azonosító ] ":" *Kifejezés* { "," *Kifejezés* ... } | "MEGNYIT" Azonosító ":" Szöveg | "LEZÁR" Azonosító |
| *Megjegyzés ::=* | "\*\*" Jel { Jel ... } |
| *Balérték ::=* | Azonosító { "[" *Kifejezés* "]" ... } |
| *Kifejezés ::=* | *Kifejezés* "ÉS" *RelációsKifejezés* | *Kifejezés* "VAGY" *RelációsKifejezés* |
| *RelációsKifejezés ::=* | *RelációsKifejezés* "=" *ÖsszegKifejezés* | *RelációsKifejezés* "/=" *ÖsszegKifejezés* | *RelációsKifejezés* "<" *ÖsszegKifejezés* | *RelációsKifejezés* "<=" *ÖsszegKifejezés* | *RelációsKifejezés* ">" *ÖsszegKifejezés* | *RelációsKifejezés* ">=" *ÖsszegKifejezés* |
| *ÖsszegKifejezés ::=* | *ÖsszegKifejezés* "+" *SzorzatKifejezés* | *ÖsszegKifejezés* "-" *SzorzatKifejezés* |
| *SzorzatKifejezés ::=* | *SzorzatKifejezés* "\*" *IndexKifejezés* | *SzorzatKifejezés* "/" *IndexKifejezés* | *SzorzatKifejezés* "DIV" *IndexKifejezés* | *SzorzatKifejezés* "MOD" *IndexKifejezés* |
| *IndexKifejezés ::=* | *IndexKifejezés* "^" *ElőtagKifejezés* | *IndexKifejezés* "@" *ElőtagKifejezés* |
| *ElőtagKifejezés ::=* | "-" *ElőtagKifejezés* | "NEM" *ElőtagKifejezés* | "SIN" *ElőtagKifejezés* | "COS" *ElőtagKifejezés* | "TAN" *ElőtagKifejezés* | "ARCSIN" *ElőtagKifejezés* | "ARCCOS" *ElőtagKifejezés* | "ARCTAN" *ElőtagKifejezés* | "LOG" *ElőtagKifejezés* | "EXP" *ElőtagKifejezés* | "RND" *ElőtagKifejezés* | "KIS" *ElőtagKifejezés* | "NAGY" *ElőtagKifejezés* | "BETŰ" *ElőtagKifejezés* | "SZÁM" *ElőtagKifejezés* | "KEREK" *ElőtagKifejezés* | "EGÉSZ" *ElőtagKifejezés* | "VALÓS" *ElőtagKifejezés* | "VÉGE" Azonosító | *AlapKifejezés* [ *KifejezésUtótag* ] | *Konstans* |
| *AlapKifejezés ::=* | Azonosító | "|" *Kifejezés* "|" | "(" *Kifejezés* ")" |
| *KifejezésUtótag ::=* | "[" *Kifejezés* "]" [ *KifejezésUtótag* ] | "[" *Kifejezés* ":" *Kifejezés* "]" |
| *Típus ::=* | "EGÉSZ" | "VALÓS" | "LOGIKAI" | "KARAKTER" | "SZÖVEG" | "BEFÁJL" | "KIFÁJL" | *Típus* "[" EgészSzám "]" |
| *Konstans ::=* | EgészSzám | Tizedestört | Szöveg | Karakter | "SV" | "IGAZ" | "HAMIS" |
| *EgészSzám ::=* | Számjegy { Számjegy ... } |
| *Tizedestört ::=* | Számjegy { Számjegy ... } "." Számjegy { Számjegy ... } |
| *Szöveg ::=* | "”" { Jel ... } "”" |
| *Karakter ::=* | "’" Jel "’" |
| *Azonosító ::=* | Betű { AzonosítóJel ... } |
| *AzonosítóJel ::=* | Betű | Számjegy |
| *Jel ::=* | Betű | Számjegy | EgyébJel |