

Dan-Hariton Ioan-Marian

Petrescu Philip Daniel

1220FB

Limbaj de DiJei

Definirea variabilelor:

În acest limbaj de DiJei variabilele se definesc ca atare :

ex : s1 = new [instrument]

« s1 » fiind numele variabilei, iar instrument fiind numele unui instrument (ex : guitar, piano)

Din moment ce o variabila este initializata cu un instrument, aceasta asimileaza toate bibliotecile de sample-uri.

Tipurile de date :

Limbajul nostru fiind unul cat se poate de simplu, atat din punct de vedere al complexitatii lui cat si al experientei de utilizare, am optat sa nu avem decat numere **intregi**, **reale** si **obiecte de tipul « instrument »**.

Structuri repetitive :

Singura structura repetitiva este « loop », aceasta iterandu-se la nesfarsit. Singura posibilitate de a iesi fortat dintr-acesta este folosirea comenziilor « **exit** » sau « **go to** », despre care vom discuta in detaliu mai tarziu.

Sintaxa structurii « loop » este urmatoarea :

ex : loop[numarul loop-ului] do :

statements

}

numarul loop-ului fiind un numar intreg (ex : loop1, loop2, etc)

Avem doua tipuri de structuri loop :

1. Structura loop simpla care nu presupune un thread individual ce o itereaza

Ex: **loop1 do:{**
 statements
 }

2. Structura thread loop care presupune un thread pentru fiecare thread-loop

Ex: **thread**
 loop1 do:{
 statements
 }

(tips: in cazul in care doriti sa folositi mai multe thread-loop-uri, evitati folosirea loop-urilor simple inaintea acestora.)

Pe langa statements-urile care se pot folosi oriunde in cod, loop-urile detin o serie de comenzi specifice :

1. **after [integer] exit**

Aceasta comanda face posibila parasirea loop-ului dupa un numar predefinit de iteratii ([integer]) .

Ex : after 100 exit

2. **after [integer] go to [name of another loop]**

Aceasta comanda face posibila parasirea loop-ului dupa un numar predefinit de iteratii, dar si redirectionarea threadului curent catre alt loop.

(tips : in momentul in care folositi aceasta comanda, este recomandat sa redirectionati threadul catre un loop simplu).

Ex : after 100 go to loop3

3. **after [integer] sleep [integer]**

Aceasta comanda apeleaza **sleep** dupa un numar predefinit de iteratii.

Ex : after 100 sleep 5

Comanda sleep are rolul de a pune pauza thread-ului pentru un anumit numar de secunde. Argumentul acestei comenzi poate avea valori reale.

Ex : *sleep 0.5*

Comanda play are rolul de a porni un sample/nota muzicala

[variabila] play [nume sample/nota muzicala]

Ex : *s1 play drums_low4*

Sintaxa comentariilor:

Comentariile sunt de forma :

comments : [text-comment] :

Ex: *comments : this is the best language :*

Exemple de cod :

S1 = new piano

S2 = new voice

S3 = new drums

thread

```
loop1 do :{  
    S1 play b4_minor_low  
    sleep 0.5  
    S1 play f4_major_low  
    sleep 1  
    after 100 go to loop 3  
}
```

thread

```
loop2 do :{  
    S3 play drums_percutie2  
    sleep 0.2
```

```

    S3 play drums_toba5
    sleep 0.1
}

loop3 do :{
    after 1 exit
    S2 play voice_over_male
    S2 play voice_over_female_high
}

```

Explicatii:

Loop 1 et loop 2 incep in acelasi timp de catre 2 threaduri. Dupa 100 de iteratii ale loop-ului 1, thread-ul se muta de pe acest loop pe loop3. In acest moment avem doar 2 loopuri care sunt iterate (1 si 3). In loop 3, threadul verifica daca s-a facut cel putin o iteratie (in acest caz, nu s-a facut niciuna). Da play la cele doua sample-uri. Dupa ce se termina cele doua sample-uri, threadul intra din nou in loop si verifica daca acesta a fost iterat macar odata (in cazul de fata, da) si iese. In final doar loop 2 este in continuare activ.

Pentru conceperea acestui limbaj de programare, ne-am inspirat din mai multe parti cum ar fi : ABC Notation (de unde am preluat faptul ca putem folosi si note muzicale pe langa sample-uri deja facute) si din Sonic Pi (am preluat ideea de thread-uri si loop-uri). De asemenea, daca ar fi sa mai lucram la acest proiect pe viitor, vom introduce biblioteci cu sample-uri deja facute de pian, chitara etc si in acelasi timp biblioteci cu note muzicale sau sunete inregistrate pentru a ne crea propriile sample-uri.