



Instituto Tecnológico de Costa Rica

Compiladores e Intérpretes IC5701

Grupo 20

Profesor:

Marco Aurelio Sanabria Rodríguez

Proyecto #1 Gramática

Ewa Káska

Estudiantes:

| | |
|-----------------------------|------------|
| Enoc Damián Castro Vargas | 2019390090 |
| Kendall Walsh Muñoz | 2018107551 |
| Francisco Gonzalez Madrigal | 2018107608 |
| Josué Huertas Mora | 2019160629 |

Tabla de Contenidos

| | |
|-----------------------|---|
| Motivación..... | 3 |
| Análisis..... | 3 |
| Aspecto chistoso..... | 3 |
| Desventajas | 3 |
| Ventajas..... | 3 |
| Gramática..... | 4 |
| Ejercicios..... | 5 |

Motivación

El lenguaje creado llamado “Ewa káska” que su traducción sería “Hola Mundo”. Dicho lenguaje, esta creado utilizando las palabras escritas en el lenguaje indígena Bribri, el cuál es un grupo étnico de Costa Rica, ubicado en Talamanca junto con los cabécares. Además, la principal diferencia de este lenguaje con respecto a otros, es que la curva de aprendizaje no está en la planeación y/o escritura del código, sino más bien de la escritura de este, ya que este al utilizar ciertos caracteres y/o palabras de la lengua indígena costarricense bribri, se tiende a perder la percepción de lo que se está queriendo escribir o de lo que se quiere hacer, por lo cual al programar durante múltiples ocasiones, es posible que el programador en cuestión nutra sus conocimientos de palabras básicas del bribri, la cual es una lengua amenazada que puede dejar de existir dentro de algunos años debido a que poco a poco hay menos hablantes, por lo cual, este lenguaje es una solución tanto tecnológica como social a esta problemática, ya que además de enseñar ciertas palabras, dará un contexto de la situación de la lengua a los programadores que la utilicen.

Análisis

Aspecto chistoso

- El lenguaje se hace gracioso en el momento que el programador se empieza a desenvolver en la sintaxis del lenguaje, ya que es divertido escuchar a las personas pronunciado las palabras en Bribri.

Desventajas

- Una desventaja del lenguaje Ewa káska es que a pesar de que sea fácil memorizar las reglas, al no conocerse el lenguaje Bribi se vuelve insípido para las personas puesto que no conocemos el significado de las palabras en el código

Ventajas

- El lenguaje tiene la ventaja de tener una sintaxis muy limpia debido a que requiere de tabulación como en Python, y además cuenta con las palabras “kéwe” y “batà”, que significan inicio y fin respectivamente, las cuales ayudar a establecer los scopes de funciones y ciclos.

Gramática

| | | |
|--|-----------------|-------------------|
| Programa ::= (Asignacion Funcion Comentario)* | | |
| Comentario ::= chōk: [A-Za-z0-9] | | |
| Asignacion ::= Identificador dōr (Valor Llamada) | | |
| Funcion ::= Identificador (Parametros) { Instrucciones + } | | |
| Identificador ::= [a-z][a-zA-Z 0-9_]+ | | |
| Valor ::= Texto Numero Booleano | | |
| Texto ::= ~ [a-z A-Z _]+~ | | |
| Numero ::= (Entero / Flotante) | | |
| Entero ::= - ? [0-9] + | | |
| Flotante ::= - ? [0-9] + . [0-9] + | | |
| Booleano ::= chōkale kōchō | | |
| Parametros ::= Identificador (* Identificador) * | | |
| Instrucciones ::= Ciclo Condicional Asignacion | | |
| Ciclo ::= amaúk Condicion { Instrucciones + } | | |
| Condicional ::= èma_ Condicion { Instrucciones + } | | |
| Condicion ::= Comparado Comparador Comparado | | |
| Comparado ::= (Identificador Valor) | | |
| Comparador ::= kibí btàie tse kuoēi | | |
| Expresion ::= (OtraExpresion) Valor Identificador | | |
| OtraExpresion ::= Expresion Operador Expresion | | |
| Operador ::= ukōfkōk skōk balātōk bērie | | |
| NOTAS | | |
| chōk decir | kibí mayor | balātōk dividir |
| dōr ser | btàie menor | bērie multiplicar |
| chōkale verdadero | tse igual | |
| kōchō falso | kuoēi diferente | |
| amaúk repetir | ukōfkōk sumar | |
| èma_si | skōk restar | |

Ejercicios

Imagen 1: Función para saber si número es primo

```
děł esPrimo(num)
ké_we
    i = num - 1
    dalèkua i > 1
        ké_we
            èta num % i == 0
                dókmǎľě kòchö
        batà
            i -= 1
        dókmǎľě chókale
batà
```

Imagen 2: Función para contar vocales en un string

```
děł contarVocales(texto)
ké_we
    vowels=0
    ie i e texto
        ké_we
            èta(i=='a' || i=='e' || i=='i' || i=='o' or i=='u' ||
                i=='A' || i=='E' || i=='I' || i=='O' or i=='U')
                vowels=vowels+1
        batà
        dókmǎľě vowels
batà
```

Imagen 3: Función para saber si un número es par

```
dēl esPar(numero)
ké_we
    ètā (numero %2 == 0)
        ké_we
            dókmalě chókale
        batà
            dókmalě kòchö
    batà
```

Imagen 4: Función para revertir una cadena de texto

```
dēl stringReverso(texto)
ké_we
    count = 0
    string_reverso = ""
    index = bakò_s(text)
    dalèkua count > 0
        ké_we
            string_reverso += texto[ count - 1]
            index = index - 1
        batà
        dókmalě string_reverso
    batà
```

Imagen 5: Función para sumar dos números

```
dēl sumar(numero1, numero2)
ké_we
    ema = numero1 + numero2
    dókmalě ema
    batà
```