






emojiのチュートリアル

1. emojiとは

emojiとはテキスト処理の講義を基に作成された非常に小さなプログラミング言語であり、手続き型言語である。シンタックスの一部に絵文字が使用されていることからemojiという名前になった。emojiは逐次実行、if文、while文、変数定義、変数呼び出し、出力が可能となっている。ただし、if文等で用いられるブロック構造はなく、if文、while文で実行される処理は同じ行に記述された1命令のみとなっている。ブロック構造がないため、関数の概念も存在していない。出力のみ可能なのは専用の文を作成し、関数としなかったためである。

2. シンタックスについて

絵文字が当てられている予約語はif文、while文、print文、println文、変数定義の5つである。それぞれの対応する絵文字は以下である。

if	
while	
print	
println	
変数定義(varなど)	

それ以外の条件演算子、算術演算子は一般的なプログラミング言語で 사용되는ものと同じである。一つ異なる点として、代入演算子は”=“ではなく”:=“である。

emojiの構文は世のプログラミング言語とほぼ同様に定義されている。ただし、文末のセミコロンは存在せず、記述するとエラーになる。改行が文末として認識されるので命令を改行して記述するとエラーするので注意が必要である。

具体的なemojiの構文は以下である。

if文	 (条件式){処理}
while文	 (条件式){処理}
print文	 (文字列, 計算式, 数値)
println文	 (文字列, 計算式, 数値)
変数定義	 変数名 := 計算式もしくは文字列, 数値
計算式	1+1, 1*1, 1-1, 1/1, (1+1)*1 など

論理式	$1 == 1, 1 != 1, 1 > 1, 1 < 1, 1 \geq 1, 1 \leq 1, 1 \parallel 1, 1 \&\& 1, !1, (1 \&\& 1) == 1$ など

変数宣言は🔒をつける以外に方法はないので、ソースコード中に宣言なしに変数を利用すると例外を飛ばすので注意が必要である。また、変数のスコープは全てグローバルスコープである。また、emojiにはブロック構造が存在しないため、