

아희

밧밧밧밧밧밧밧따따따따따따따밧밧밧밧따따뚜
븧븧븧븧븧뎡뎡뎡뎡뎡뎡뎡뎡븧븧븧븧븧뎡뎡
따따따따밧밧밧따따밧밧반따따뵤다다다다망희

아희

한글로

쓰는 난해한

프로그래밍 언어

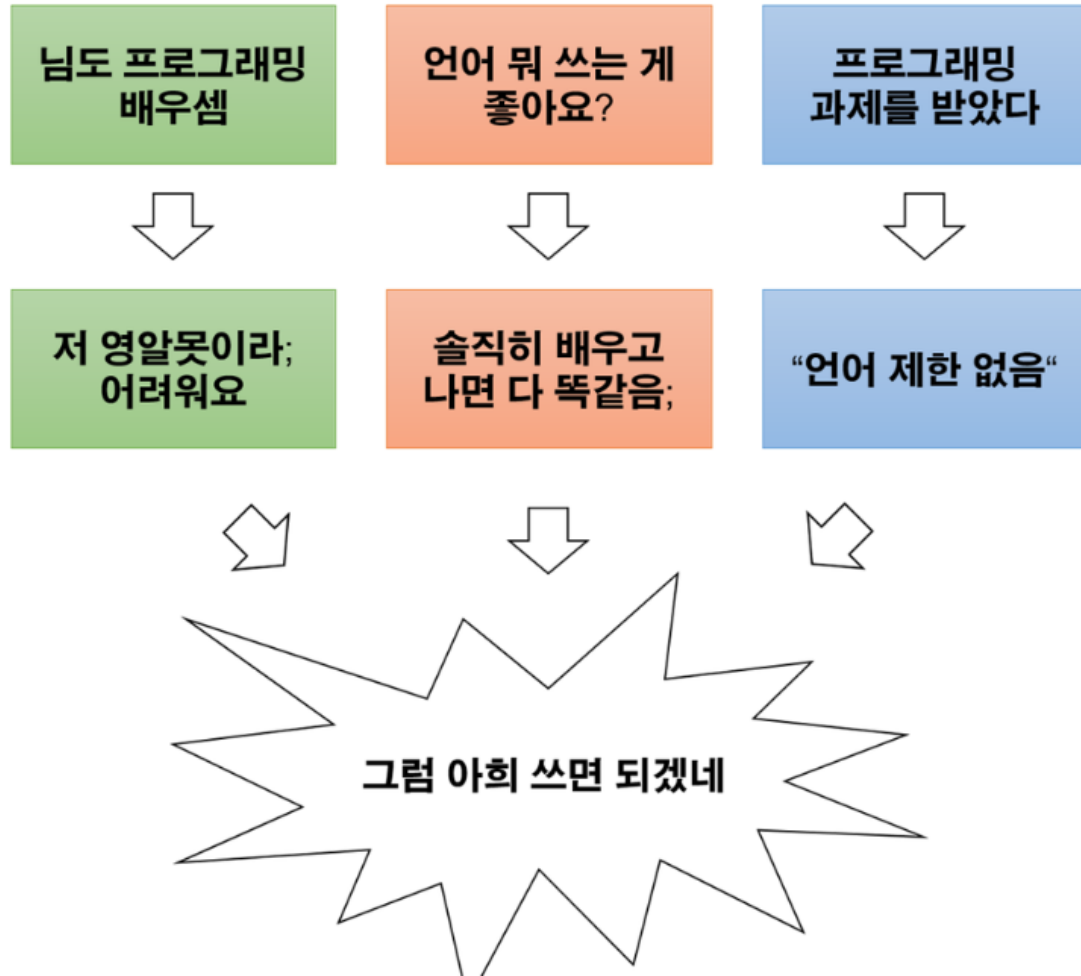
아희란?

- '아이'를 뜻하는 고어
- 한글로 된 프로그래밍 언어 중 가장 먼저 발표된 난해한 프로그래밍 언어
- 저장공간으로 26개의 '스택'과 1개의 '큐'(○)를 사용
- 2차원 언어
- 엔하위키 미리 개발자인 Puzzlet Chung가 개발
- JavaScript로 구현한 아희 컴파일러 http://puzzlet.org/doc/aheui/jsaheui_ko.html
- String to Aheui <https://apteryx.moe/straheui/>
- 아희 소개 <https://aheui.readthedocs.io/ko/latest/index.html>
- 국제 아희 개발자 대회 페이스북 <https://www.facebook.com/aheuicon>

아희를 만든 이유

- 1. 심심해서
- 2. 심심해서
- 3. 심심해서
- 4. 한글 낱자의 디자인을 살릴 수 있는 프로그래밍 언어를 개발하기 위해 (?)

아희를 쓰는 이유



아희의 위대함

- 한글의 우수함을 그대로 반영한 유일한 프로그래밍 언어
- 같은 계열의 자모는 비슷한 소리를 내는 것에서 착안
 - ★ 같은 계열의 자모는 비슷한 기능을 수행!
- 'ㄷ', 'ㅌ', 'ㄸ', 'ㄴ', 'ㄹ' => 연산 '+', '-', '*', '/', '%'
- 'ㅇ', 'ㅂ' => 뽑기, 집어넣기
- 'ㅏ', 'ㅑ', 'ㅓ', 'ㅕ', 'ㅗ', 'ㅛ', 'ㅜ', 'ㅠ' => 속력과 방향을 쉽게 유추가능(상형)
- 적은 byte로 원하는 프로그램을 제작가능
 - ex) 뺑망희(문자->아스키코드값), 뺑뺑다망해(A+B)
- 개인의 창의성을 발휘하기 최적화된 언어!
- 이러한 아희의 위대함을 인정받아 백준에서도 아희코드 제출이 가능함!

A+B

언어

아희

소스 코드 공개

☒ 공개

☐ 비공개

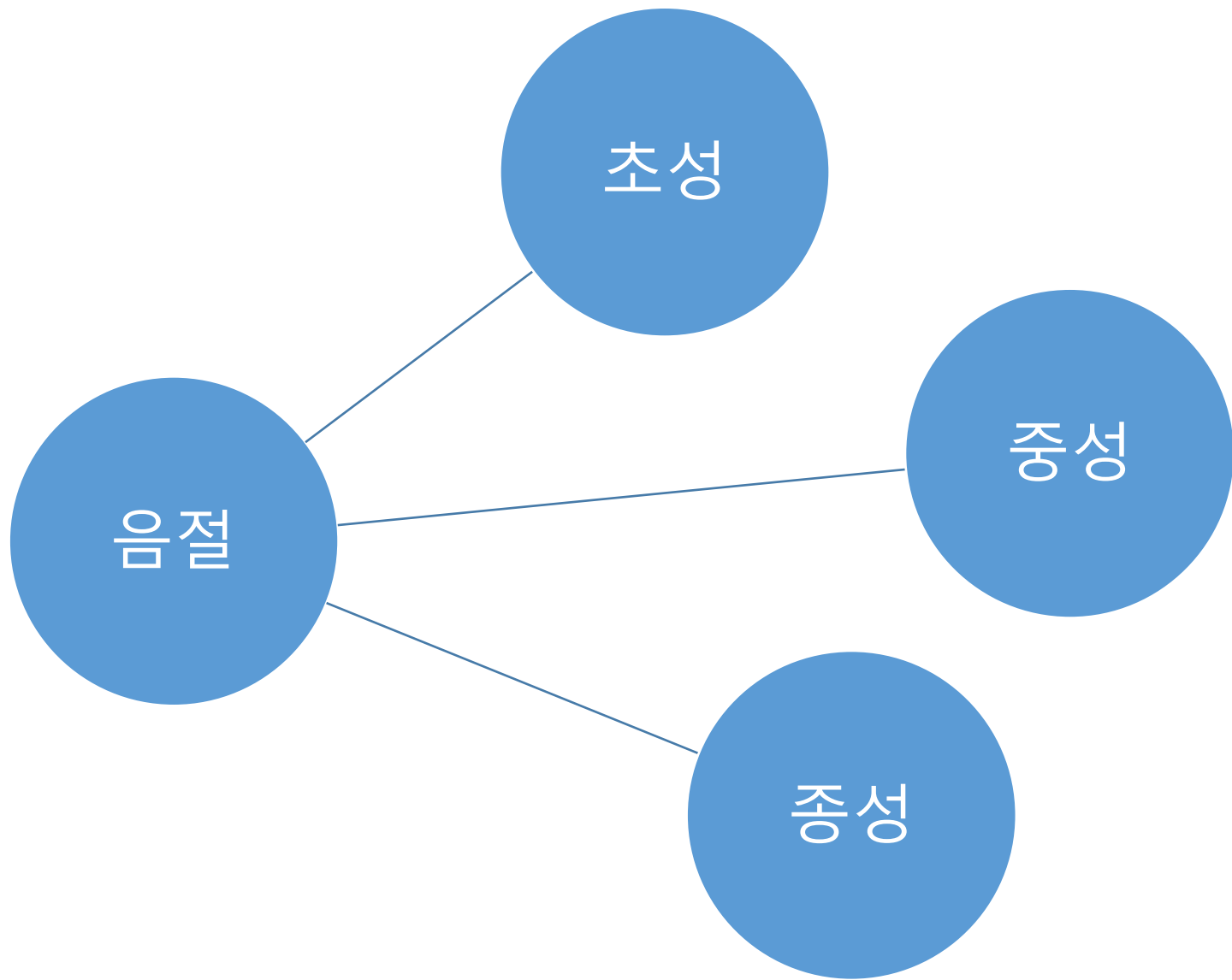
☐ 맞았을 때만 공개

소스 코드

1 방방다망희

채점 번호	아이디	문제 번호	결과	메모리	시간	언어	코드 길이	제출한 시간
13051822	suhan8984	1000	맞았습니다!!	31268 KB	4 ms	아희 / 수정	15 B	38초 전

문법



문법 - 초성

ㄱ, ㅇ 묶음

- ㅇ, ㄱ, ㅋ: 아무 일도 하지 않는다.^[6]
- ㅎ: 프로그램을 종료한다. 종료 시 선택된 저장 공간에 값이 하나 이상 남아 있다면 하나만 뽑아내 운영체제에 반환하고 비어있다면 0을 반환한다.

◦ ㄷ 묶음 - 셈

- ㄷ, ㄸ, ㅌ, ㄴ, ㄹ: 각각 덧셈, 곱셈, 뺄셈, 나눗셈, 나머지 연산(modulo).
- 모두 공통적으로 저장공간에서 두 값을 뽑아낸 다음 둘을 연산한 값을 저장공간에 저장한다. 순서가 있는 연산자의 경우 나중에 꺼낸 값이 연산자의 왼쪽에 온다.

◦ □ 묶음 - 저장공간

- □: 뽑기. 지금 저장공간에서 자료 구조의 형태에 맞게 하나 꺼내온다. 종성으로 ㅇ이 오면 숫자로, ㅎ이 오면 문자로 출력하고 나머지 종성이면 뽑아내서 그냥 버린다.
- ㅂ: 집어넣기. 지금 저장공간에 값을 하나 넣는다. 종성이 ㅇ, ㅎ이면 각각 숫자와 문자로 사용자에게 입력을 받고, 나머지는 글자의 선의 수가 집어넣을 값이 된다. ㅅ→2, ㄹ→5,... 이런 식으로. 1을 곧바로 넣을 수가 없는데, 3 - 2(받반타, 박반타 등)를 하면 된다. 종성이 없으면(바, 보 등) 0을 넣는다.
- ㅃ: 중복. 지금 저장공간이 스택이라면 맨 위의 값을 그 값 위에 하나 더 집어넣고, 큐라면 맨 앞의 값을 앞에 하나 더 덧붙인다. 통로라면 아까 보낸 값을 한 번 더 보낸다.
- ㅍ: 바꿔치기. 지금 저장공간이 스택이라면 맨 위 값과 그 바로 아래 값을, 큐라면 맨 앞의 값과 그 바로 뒤 값을 바꾼다. 통로일 때의 동작은 미정.

◦ ㅅ 묶음 - 제어, 저장공간 확장

- ㅅ: 선택. 종성을 통해 상술한 자료구조에서 선택 가능.
- ㅆ: 이동. 지금 저장공간에서 값 하나를 뽑아내서 받침이 나타내는 저장공간에 그 값을 집어넣는다.
- ㅈ: 비교. 저장공간에서 값 두 개를 뽑아 내서 비교한다. 나중에 뽑아낸 값이 더 크거나 같으면 1을, 아니면 0을 지금 저장공간에 집어넣는다.
- ㅊ: 조건. 저장공간에서 값 하나를 뽑아내서 그 값이 0이 아니면 진행해야 할 방향으로, 0이면 그 반대 방향으로 간다.

중성

- ㅏ, ㅑ, ㅓ, ㅕ: 중성의 방향이 커서의 방향이 된다. 각각 우,좌,상,하.
 - ㅗ, ㅛ, ㅛ, ㅜ: 위와 같으나 2칸 옮긴다.
 - ㅣ, ㅡ, ㄴ: ㅣ는 커서가 세로 방향으로 들어오면 그대로 통과, 가로로 돌아오면 전 자리로 옮기고 방향을 뒤집는다. ㅡ도 가로가 통과이고 세로가 뒤집을 뿐이지 ㅣ와 같다. ㄴ는 어느 방향으로 들어오든 방향을 뒤집는다.
 - 기능 없음: ㅈ ㅊ ㅊ ㅊ ㅊ ㅊ ㅊ ㅊ ㅊ ㅊ (커서가 이동하는 방향과 크기는 변하지 않습니다.)
- 한글이 아닌 모든 글자는 빈칸으로 처리된다. 즉 무시된다.

Hello, World! – 아희 사이트 코드

복습

- 아희 코드는 왼쪽 위부터 시작
- 종성(ㅏ, ㅑ, ㅓ, ㅕ)은 방향을 지정
- ㅂ: push => 종성의 '선의 수'를 push
- ㅄ: duplicate
- ㅍ: pop => 종성 ㅇ: 숫자, ㅎ: 문자
- ㅌ: plus
- ㅎ: multiply
- ㅎ: end

밤밭따빠밭밧따뿌
빠말파빨받밤뚜물
돈박탕빠말붓두분
본몰박밭똥두물붓
몰도몰희멀물물복
몰복토펴더벌뿌뚜
뽕뽕멀멀더벌배돈
뽕딩벌멀배덕더벅

실행 순서

아 히
한 글 로
쓰 는 난 해 한
프 로 그 래 밍 언 어



밤밤따빠밤밤따부
 빠말과빨반밤부
 돈박탕빠말부
 돈돈박말부
 돈돈히말부
 돈돈도범더벌부
 뽕뽕말말더벌빠돈
 뽕뽕벌말빠덕더벌

ind	code	meaning	stack	comment
0	밤	push	[4]	
1	밤	push	[4, 8]	
2	따	*	[32]	
3	빠	dup	[32, 32]	
4	밤	push	[32, 32, 8]	
5	밤	push	[32, 32, 8, 9]	
6	따	*	[32, 32, 72]	
7	부	dup	[32, 32, 72, 72]	
8	뽕	pop	[32, 32, 72]	H 출력
9	분	push	[32, 32, 72, 3]	
10	붓	push	[32, 32, 72, 3, 3]	

밤발따빠발뽕따부
 빠말파빨반밤두뽕
 돈박탕빠말뽕두돈
 뽕뽕박발뽕두뽕뽕
 뽕도뽕희뽕뽕뽕뽕
 뽕뽕도뽕뽕뽕뽕뽕
 뽕뽕뽕뽕뽕뽕뽕뽕
 뽕뽕뽕뽕뽕뽕뽕뽕

11	꺾	push	[32, 32, 72, 3, 3, 3]	
12	두	*	[32, 32, 72, 3, 9]	
13	돈	*	[32, 32, 72, 27]	↳ 무시 : 산술 연산의 받침은 무시
14	뽕	push	[32, 32, 72, 27, 2]	
15	더	+	[32, 32, 72, 29]	
16	덕	+	[32, 32, 101]	
17	빠	dup	[32, 32, 101, 101]	
18	뽕	pop	[32, 32, 101]	e 출력
19	벗	push	[32, 32, 101, 7]	
20	더	+	[32, 32, 108]	
21	뽕	dup	[32, 32, 108, 108]	
22	뽕	dup	[32, 32, 108, 108, 108]	
23	뽕	pop	[32, 32, 108, 108]	l 출력

밤발따빠발뽕따뿌
 빠말파빨반밤두뽕
 돈박탕빠말뽕두돈
 뽕뽕박발뽕두뽕뽕
 뽕도뽕희말뽕뽕뽕
 뽕뽕도뽕뽕뽕뽕
 뽕뽕말말뽕뽕빠돈
 뽕뽕뽕말빠뽕뽕뽕

24	뽕	pop	[32, 32, 108]	I 출력
25	본	push	[32, 32, 108, 3]	
26	돈	+	[32, 32, 111]	
27	빠	dup	[32, 32, 111, 111]	
28	말	pop	[32, 32, 111]	O 출력
29	파	swap	[32, 111, 32]	
30	빨	dup	[32, 111, 32, 32]	
31	반	push	[32, 111, 32, 32, 3]	
32	밤	push	[32, 111, 32, 32, 3, 4]	
33	두	*	[32, 111, 32, 32, 12]	
34	두	+	[32, 111, 32, 44]	
35	뽕	pop	[32, 111, 32]	, 출력
36	뽕	pop	[32, 111]	space 출력
37	뿌	dup	[32, 111, 111]	
38	빠	dup	[32, 111, 111, 111]	

밤발따빠발뽕따부
 빠말파빨반밤두뽕
 돈박탕빠말뽕두돈
 돈뽕박발뽕투뽕뽕
 뽕도뽕희말뽕뽕뽕
 뽕뽕도범더벌부두
 뽕뽕말말더벌빠돈
 뽕뽕벌말빠덕더뽕

39	뽕	push	[32, 111, 111, 111, 8]	
40	더	+	[32, 111, 111, 119]	
41	말	pop	[32, 111, 111]	w 출력
42	말	pop	[32, 111]	o 출력
43	뽕	dup	[32, 111, 111]	
44	뽕	push	[32, 111, 111, 3]	
45	도	+	[32, 111, 114]	
46	뽕	pop	[32, 111]	r 출력
47	박	push	[32, 111, 3]	
48	탕	-	[32, 108]	
49	빠	dup	[32, 108, 108]	
50	말	pop	[32, 108]	l 출력
51	뽕	push	[32, 108, 8]	
52	투	-	[32, 100]	
53	뽕	pop	[32]	d 출력

밤발따빠발뽕따뿌
 빠말파빨반밤두뽕
 돈박탕빠말뽕두뽕
 뽕뽕박발뽕두뽕뽕
 뽕노뽕히말뽕뽕뽕
 뽕뽕뽕뽕뽕뽕뽕
 뽕뽕뽕뽕뽕뽕뽕
 뽕뽕뽕뽕뽕뽕뽕
 뽕뽕뽕뽕뽕뽕뽕

54	뽕	push	[32, 5]	
55	더	+	[37]	
56	뽕	push	[37, 4]	
57	토	-	[33]	
58	뽕	pop	[]	! 출력
59	박	push	[2]	
60	발	push	[2, 5]	
61	뽕	*	[10]	
62	뽕	pop	[]	linefeed (10) 출력
63	히	end	[]	ㅎ 은 끝

Hello, World! – 나무위키 코드

16 15 14 13
발 뽀 뽀 뽀 더 뽀 뽀
더 뽀 뽀 더 뽀 뽀 뽀 뽀
나 파 파 뽀 뽀 뽀 뽀 뽀 뽀
뽀 뽀 뽀 뽀 뽀 뽀 뽀 뽀
더 뽀 뽀 뽀 뽀 뽀 뽀 뽀

- 출력 결과가 같지만 8x8에서 8x5로 무려 24자나 줄어들었다!!!

유명한 예제

- 아희 : 프로그램 종료
- 방망희 : 입력을 받아 숫자로 출력해주는 코드
- 발받악에땀망 : 숫자 15를 출력해주는 코드

기초 예제) 아래의 코드를 실행시키면?

분해여
뿌빠뿌
빠뽀뿡

$A+B$, $A-B$, $A*B$, A/B

- 방방다망해
- 방방타망해
- 방방따망해
- 방방나망해

=> **아희는 간결하다!**

'ㅌ'을 이용한 반복문

- 삭받받타사붕
- 투터번벌씩빠
- 삭다사빠아초
- 파플짱희멍석
- 밧받타방뿌너백퍼더
- 백힝멍머차파받밧토

예제1) 1부터 N까지 합

삭받받타 : 스택 'ㄱ'에 0을 넣음

사붕: 숫자를 입력받아 스택 ' '에 넣음

빠씩: 스택 ' '의 top에 존재하는 숫자를 복사해서 스택 'ㄱ'에 넣음

벌번터: 스택 ' '에 1을 넣음

투: 스택 ' '에서 x-1연산을 수행

삭다: 스택 'ㄱ'에서 더하기 연산을 수행

사빠아초: 스택 ' '의 top에 있는 수가 0인지 아닌지 확인

석멍희: 스택 'ㄱ'에 존재하는 수를 출력한 후 프로그램종료

짱플파: 실행되지 않음

예제2) $\log_2(N)$ 의 근사

관심있는 사람은 분석해보기(절대 귀찮아서 아님)

아희아희 - 인터프리터

아희아희 • 창작배경

2015년 4월 1일

일찍이 세종대왕께서는 못 백성을 가엽게 여겨 과학적인 문자 한글을 창제하여 널리 쓰이도록 하였습니다. 이 훌륭한 전통을 이어내려, 지난 이천오년에 퍼즐릿 정의 주도로, 못 개발자들을 가엽게 여긴 뜻 있는 사람들이 모여 한글 프로그래밍 언어를 만들어 널리 쓰이도록 하였으니, 이것이 바로 아희입니다. 아마 많이 들어 보셨을 것입니다.

지난 열 해 동안 아희는 방망희질을 하는데 널리 쓰이고 그 유용성을 인정받았으며, 못 학생들의 창의력을 증진시켰고, 아희의 우수성에 감화받은 노르웨이 개발자들은 헌정 음악을 지어 아희를 널리 알리기도 하였습니다. 그러나 이 모든 일에도 불구하고 아희는 그동안 다른 언어로 구현되어야만 했으며, 지난 열 해동안 많은 아희 개발자들의 아희질에도 불구하고 우리 글자로 우리 프로그래밍을 하려는 꿈은 이루어지지 못 했습니다.

그러나 그것도 이제 옛 말입니다. 국제 아희 개발자 기구는 오는 568½돌 한글날을 맞이하여 아희아희를 소개합니다.

[illegible]

알파히 - 알파히썬(rpython)으로 만든 엔터프라이즈급 고성능 아히

build failing

- English: [README.en.md](#)
- Working log (English): [LOG.md](#)
- 빌드 및 실행 영상: [Youtube](#)
- 2015 한국 파이콘: [PyPy/RPython으로 20배 빨라지는 JIT 아히 인터프리터](#)

알파히썬(RPython)은 PyPy를 개발하기 위해 개발/사용된 python의 방언으로서 정적 언어로 컴파일되고 tracing-JIT를 지원하기 위한 라이브러리를 내장하고 있습니다. 알파히썬으로 개발한 언어는 손쉽게 파이썬으로 실행하거나 바이너리로 빌드할 수 있고, JIT를 적용하기도 쉽습니다.

이 프로젝트는 RPython으로 JIT 인터프리터를 개발하는 예제로 활용할 수 있습니다. 위의 링크에서 LOG.md를 확인해 더 알아보세요.

밥밧밧밧밧밧밤따따따따따밤밧밧우
우어어어어어어어어어어어어어어
밧밧따따따따밧밧반따따밧다다다망우
우어어어어어어어어어어어어어어어
밥밧밧밧밧밧밤따따따따따밧밧밧우
우어어어어어어어어어어어어어어어
밧밧따따따따밧밧밧밤따따따박밧반우
우어어어어어어어어어어어어어어어
따따밥다다다다망밧밧밧밧밧밤따우
우어어어어어어어어어어어어어어어
따따따따따밥밧밧밧밧따따따밧밧우
우어어어어어어어어어어어어어어어
밧밤따따따밧밧반따따받다다다망우
우어어어어어어어어어어어어어어어
밥밧밧밧밧밧밤따따따따따밧밧밧우
우어어어어어어어어어어어어어어어
밧밧따따따따밧밧밧밤따따따밧반우
우어어어어어어어어어어어어어어어
따따밧다다다다망밧밧밧밧밧밤따우
우어어어어어어어어어어어어어어어
따따따따따밧밧밧밧따따따밧밧우
우어어어어어어어어어어어어어어어
밧밤따따따밧밧반따따밧다다다망우
우어어어어어어어어어어어어어어어
받밧반따따받다망희

아희
한글로
쓰는난해한
프로그래밍언어