



การเขียนโปรแกรมพื้นฐาน Fundamental Programming

- วากยสัมพันธ์ และความหมายของภาษาระดับสูง
- ตัวแปร ชนิด นิพจน์ และการกำหนดค่า

การกำหนดวากยสัมพันธ์ Syntax Definition

- การกำหนดวากยสัมพันธ์ทำได้โดยกำหนดกฎ(rule) ต่างๆ หรือ production เพื่อสร้าง nonterminal symbol จาก
 - Nonterminal symbol อื่นๆ และ/หรือ
 - สัญลักษณ์ท้าย
- ตัวกำหนดวากยสัมพันธ์ของภาษา = เซตของ nonterminal symbol + เซตของ production



วากยสัมพันธ์ของภาษาระดับสูง Syntax of a higher-level languages

- Backus-Naur Form หรือ BNF
- สัญลักษณ์ท้าย(Terminal symbol) หมายถึง สัญลักษณ์พื้นฐาน หรือเรียกอีกอย่างว่า โทเคน (Token) ได้แก่ ชื่อของข้อมูล เครื่องหมายวรรคตอน และคำเฉพาะ เช่น if, while, return เป็นต้น
- การผสมสัญลักษณ์เข้าด้วยกันเกิดเป็นโครงสร้างที่ซับซ้อนขึ้น เรียกโครงสร้างเหล่านี้ว่า Nonterminal symbol หรือ syntactic category ของภาษา



ตัวอย่างการกำหนดวากยสัมพันธ์

รายการ \rightarrow ชื่อบุคคล ที่อยู่ หมายเลข
 ชื่อบุคคล \rightarrow ชื่อต้น ชื่อสกุล { ชื่อสกุล }
 ชื่อสกุล \rightarrow ชื่อ
 ชื่อต้น \rightarrow ชื่อเฉพาะ | ตัวย่อ
 ชื่อเฉพาะ \rightarrow ชื่อ
 ตัวย่อ \rightarrow พยัญชนะ .
 ที่อยู่ \rightarrow หมายเลข ชื่อถนน
 ชื่อถนน \rightarrow ชื่อ
 พยัญชนะ \rightarrow ก | ข | ค | ... | ฮ



- การเลือกดำเนินการตามเงื่อนไข (Conditional)
- การทำซ้ำ (Loop)
- การกำหนดค่า (Assignment)

$relations \rightarrow = | \neq$

SET X TO X - Y

ความหมายของภาษาระดับสูง

- Terminal symbol และ Nonterminal symbol ของภาษาระดับสูงจะต้องถูกกำหนดให้มีความหมาย เกิดเป็นองค์ประกอบของภาษานั้น เช่น ตัวอักขระ ค่าคงที่ ตัวแปร นิพจน์ และ คำสั่ง เป็นต้น

ตัวอย่างวากยสัมพันธ์ของภาษาซี

```
for ([expression-1];[expression-2];[expression-3]) statement
    ;
```

องค์ประกอบของภาษาซี

Elements of C languages

สิ่งที่ก่อให้เกิดเป็นคำสั่งและระเบียบการใช้คำสั่ง

- ชุดของอักขระ
- ค่าคงที่
- คำหลักหรือคำสงวน
- ตัวแปร
- ตัวดำเนินการ
- นิพจน์
- คำสั่ง

Character Set ชุดของตัวอักษร

- ตัวพิมพ์เล็ก (Lowercase letters)
a b c d e ... x y z
- ตัวพิมพ์ใหญ่ (Uppercase letters)
A B C D E ... X Y Z
- ตัวเลข (Digits)
0 1 2 3 4 ... 7 8 9
- สัญลักษณ์อื่นๆ (Other characters)
+ - * / \ = () { } [] < > ! ; : ? ' " # , \$ % ^
& _ | ~ .(dot)
- อักขระพิเศษ (White space characters)
Blank, new line, carriage return, form feed, line feed,
tab, vertical tab

9

Constants ค่าคงที่

- ค่าคงที่เลขจำนวนเต็ม (Integer constant)
- ค่าคงที่เลขมีจุดทศนิยม(Floating-point constants)
- ค่าคงที่อักขระ (Character constants)
- ค่าคงที่สายอักขระ (String constants)

11

Escape sequences รหัสควบคุม

- \a Alert (bell)
- \b Backspace
- \f Form feed
- \n New line
- \r Carriage return
- \t Tab
- \v Vertical tab
- \0 Null character
- \' Single quotation mark
- \" Double quotation mark
- \\ Back slash
- \ddd ASCII character in octal notation
- \xdd ASCII character in hexadecimal notation

10

Integer constant ค่าคงที่เลขจำนวนเต็ม

- รูปแบบ (Syntax)
digits | 0*odigits* | 0x*hdigits* | 0X*hdigits*

- ตัวอย่าง

Decimal constants	Octal constants	Hexadecimal constants
10	012	0xa หรือ 0XA
32719	077717	0x7fcf หรือ 0X7FCF
78L	0116L	0x4eL หรือ 0X4EL
32768U	0100000U	0x8000U หรือ 0X8000U
65537UL	0200001U	0x10001UL หรือ 0X10001UL

12

ค่าคงที่เลขมีจุดทศนิยม Floating-point constant

■ รูปแบบ (Syntax)

digits. | *.digits[E|e [-|+]digits]* |
digitsE|e [-|+]digits | *digits.digits[E|e [-|+]digits]*

■ ตัวอย่าง

15.75	1.575e1	1575e-2
.75	75.	.0075e2
1.23	1.23L	25E-4

13

Character constant ค่าคงที่อักขระ

■ รูปแบบ (Syntax)

'อักขระ' | 'รหัสควบคุม'

■ ตัวอย่าง

ค่าคงที่	ค่า หรือความหมาย
'a'	ตัวอักษร a
'\a'	Alert หรือเสียงบี๊บ
'\\'	เครื่องหมาย \
'\x1B'	ตัวอักขระ Esc

14

String constant ค่าคงที่สายอักขระ

■ รูปแบบ (Syntax)

"ข้อความ" ["ข้อความ"]

■ ตัวอย่าง

"This is a string literal."

This is a string literal. \0

"Hi" "My students"

HiMy students \0

"A" ≠ 'A'

"" เรียกว่า Null String คุณคิดว่าสายอักขระนี้มีขนาดเท่าใด?

15

แข่งกันหน่อย.....มั๊ย?

- ♣ แบ่งผู้เรียนออกเป็น 2 ทีม
- ♣ ใช้ White board คั่นกลางระหว่างทีม
- ♣ ให้แต่ละทีม เขียนตัวอย่างค่าคงที่ ในภาษา C ครบทุกประเภท จำนวน 20 ตัวอย่าง
- ♣ ให้ฝ่ายตรงข้าม ทำวงกลมล้อม ค่าคงที่ ที่ถูกต้อง
- ♣ ร่วมกันตรวจผล พร้อมชี้แจงเหตุผล 🤔

CS112

16

Keyword คำหลัก หรือ คำสงวน

auto	double	int	struct
break	else	long	switch
case	enum	register	typedef
char	extern return	union	
const	float	short	unsigned
continue	for	signed	void
default	goto	sizeof volatile	
do	if	static	while

The additional keywords in Turbo C

asm cdecl far huge interrupt near pascal

17

ตัวแปร

การประกาศตัวแปร

<ชนิดข้อมูล> <ชื่อตัวแปร> [=ค่าเริ่มต้น][, ตัวแปร[=ค่าเริ่มต้น]];

ชนิดข้อมูล

char	signed char	unsigned char
short	int	long
unsigned short	unsigned	unsigned long
float	double	long double

ตัวอย่าง

```
char    c, msg[] = "Your welcome!";
int     i,  x = 5;
unsigned u=32768;
```

19

Variables ตัวแปร

ตัวแปร ประกอบด้วย 2 ส่วนสำคัญ

- ชื่อ
 - ข้อมูล ซึ่งเรียกว่าเป็น ค่าของตัวแปร
 - ชื่อตัวแปร ← ตัวอักษร ตัวเลข _ (เครื่องหมาย underscore)
 - กฎการตั้งชื่อตัวแปร
 - ไม่ตรงหรือซ้ำกับคำหลัก
 - อักขระตัวแรกของชื่อต้องเป็นตัวอักษร หรือ _ เท่านั้น
 - ห้ามใช้อักขระอื่น นอกจาก ตัวอักษร ตัวเลข และ _ เป็นส่วนหนึ่งของชื่อตัวแปร
 - ความยาวของชื่อไม่จำกัด แต่ถือเพียง 31 ตัวแรกเป็นนัยสำคัญ
 - acc _ok S_date addr1 ChiangMaiUniversity
- *Don't #account char 707james

18

ตัวดำเนินการ (Operators)

- ตัวดำเนินการเชิงคำนวณ (Arithmetic operators)
- ตัวดำเนินการเพื่อกำหนดค่า (Assignment operators)
- ตัวดำเนินการเปรียบเทียบ (Relational operators)
- ตัวดำเนินการตรรกะ (Logical operators)
- ตัวดำเนินการพิเศษ (Special operators)
 - Increment operator ++
 - Decrement operator --
 - Address operator &
 - Indirection operator *
 - Conditional operator ? :

20

Arithmetic Operators

สัญลักษณ์	ความหมาย	ตัวอย่าง
-	Negation	-x
*	คูณ	x * 2
/	หาร	x / 2
%	หารเก็บเศษ	x % 2
+	บวก	x + 2
-	ลบ	x - 2

21

Arithmetic operators

การประกาศตัวแปรและกำหนดค่าเริ่มต้น

```
int a=1, b=2, c=3, d=4;
```

นิพจน์	นิพจน์เทียบเท่า	ค่าของนิพจน์
a * b / c	(a * b) / c	0
a + b % c * 1	a + ((b % c) * 1)	3
8 - -d / b + c	(8 - ((-d) / b)) + c	13

23

ลำดับความสำคัญของตัวดำเนินการ

ตัวดำเนินการ
ดำเนินการ

ลำดับ

()
+ (unary) - (unary)
* / %
+ -

ซ้ายไปขวา

ขวาไปซ้าย

ซ้ายไปขวา

ซ้ายไปขวา

22

Assignment Operators

■ รูปแบบ

■ ตัวแปร = นิพจน์

■ ตัวแปร op= นิพจน์

ตัวแปร = (ตัวแปร op (นิพจน์))

■ สัญลักษณ์

ตัวอย่าง

ความหมาย

=	Simple assignment	x = 2	x = 2
*=	Addition assignment	x *= 2	x = x*2
/=	Division assignment	x /= 2	x = x/2
%=	Remainder assignment	x %= 2	x = x%2
+=	Addition assignment	x += 2	x = x+2
-=	Subtraction assignment	x -= 2	x = x-2

24

ลำดับความสำคัญของตัวดำเนินการ

ตัวดำเนินการ ดำเนินการ

ลำดับ

()
+ (unary) -(unary)
* / %
+ -
= += -= *= /= %=

ซ้ายไปขวา
ขวาไปซ้าย
ซ้ายไปขวา
ซ้ายไปขวา
ขวาไปซ้าย



25

Math library

■ #include <stdlib.h>

■ Function prototype

description

double	sqrt (double)	หาค่ารากที่สอง
double	exp (double)	หาค่า e^x
double	log (double)	หาค่า log ฐาน e
double	sin (double)	Sine
double	cos (double)	Cosine
double	tan (double)	Tangent
double	pow (double, double)	หาค่า x^y
int	rand ()	หาค่าเลขสุ่ม
int	abs(int)	หาค่าสัมบูรณ์

27

Assignment operators

การประกาศตัวแปรและกำหนดค่าเริ่มต้น

int a=1, b=2, c=3, d=4;

นิพจน์ นิพจน์เทียบเท่า ค่าของนิพจน์

a += b + c	a = (a + (b + c))	6
b *= c = d + 5	b = (b * (c = (d + 5)))	18

26

นิพจน์

■ รูปแบบ

[ค่าคงที่] [ตัวดำเนินการ] ค่าคงที่
[ค่าคงที่] [ตัวดำเนินการ] ตัวแปร
ตัวแปร ตัวดำเนินการ ค่าคงที่
ตัวแปร ตัวดำเนินการ ตัวแปร

■ ชนิดข้อมูล

ชนิดข้อมูลของนิพจน์ขึ้นอยู่กับชนิดข้อมูลของค่าคงที่ หรือตัวแปรในนิพจน์ หากค่าคงที่หรือตัวแปรมีชนิดข้อมูลต่างกัน ค่าของนิพจน์จะมีชนิดข้อมูลตามชนิดที่มีขนาดใหญ่

■ ขนาดของชนิดข้อมูล

char < short < int < long < float < double < long double

28



จงพิจารณาตัวอย่างโปรแกรมนี้ แล้ว แจ้งว่ามีองค์ประกอบของภาษาซี ประเภท ใดบ้าง?

```
#include <stdio.h>
int main() {
    int  a = 1, b = 2,  c = 3;
    x = a + b;
    print("x = %d\n", x);
    return 0;
}
```

เมื่อทำการ compile โปรแกรมนี้จะมีข้อผิดพลาดปรากฏดังนี้

