

9 Control Flow - Assignment

Thursday, October 15, 2020 8:35 AM

* Lab 3 is posted (due: 10/22)
 * Assignment 3 is posted (10/26)
 * Read chapter 6 from Scott.

var = expression ✓
 var := expression ✓
 var ← expression

l-value = &a
 r-value

Assignment Expressions

var = expr is itself an expression
 value

x = a = b = 4;
 4

Watch out!

C, C++ handle boolean values

0 - false
 non-zero - true

if (a = b) { // a == b
 |
 }
 syntax error

In Python

- Any string other than "" is true
- Any int - except 0 is true
- Any list, tuple, or dictionary is true except [], (), {}, None

if (a < b < c) {
 |
 }
 0/2

C
 • no compiler error

Valid in Python

a < b < c ≡ a < b and b < c

9 Control Flow - Assignment

Thursday, October 15, 2020 8:35 AM

Combination Assignment Operators

`+=, -=, *=, /=, %=` C, C++, Java, Python, Go



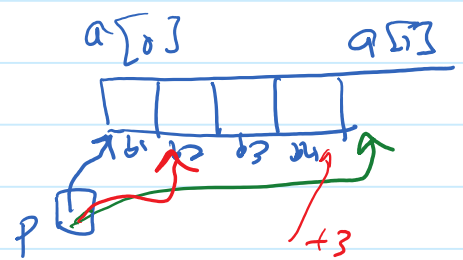
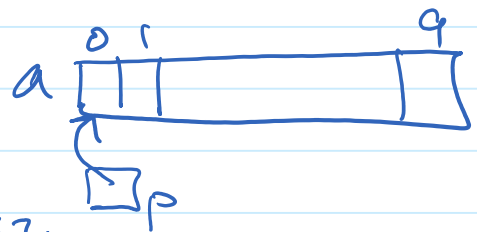
Pre/post increment/decrement operators

`++, --` C, C++, Java [Not Python]

$++i, i++ = i = i + 1$
 $a = b + ++i; \leftarrow$
 $a = b + i++;$

Pointer variables e.g. C.

int a[10];
int *p;
 $p = a; \quad p = \&a[0];$
 $p++;$ $p += 3;$
? ?



Multiway Assignment

Python
Go

$width, height = 500, 600$

$var_1, var_2, \dots = e_1, e_2, \dots$



$x, y = y, x$

$found, index = search(a, item)$

Fns can return multiple values

Variable initializations: Read in text. | Zero-initializations

9 Control Flow - Sequencing

Thursday, October 15, 2020 8:35 AM

All instructions are carried out in the sequence written:

Do this
Then do this
And then this
etc.

sequencing

→ Unstructured Flow

Fortran

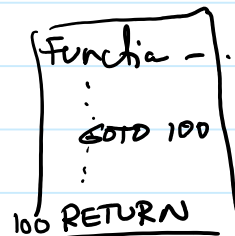
```
10  if (a .LE. b)  GOTO 25
15  min = b
20  GOTO 30
25  min = a
30  - - - -
```



"GOTO statements considered harmful"

spaghetti code

- structured statements - if, loops
- exit from a loop: **break, continue**
- returning from function calls - **return**
- returning/exiting from a deeply nested function call.
exceptions + exception handling.



Structured Flow/Programming

Essentials of Structured Control Flow

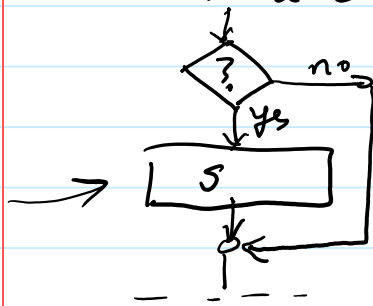
- sequencing
 - selection
 - iteration
- } + functions
- } → + Abstractions

9 Control Flow - Selection

Thursday, October 15, 2020 8:35 AM

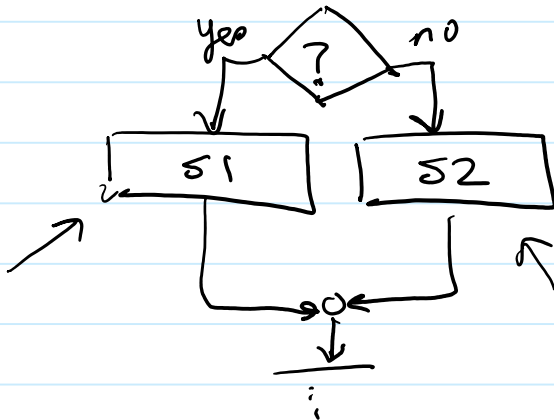
Allows a choice among a set of statements

①



```
if <condition> then
  begin
    S
  end
```

②



```
if <condition> then
  begin
    S1
  end
else
  begin
    S2
  end
```

Pascal

Designs

```
if (<c>) {
  S1;
}
else {
  <S2>;
}
```

optional
C, C++, Java

Python

```
if <c>:
  S1
else:
  S2
```

Lisp

```
(cond (<c1> <e1>)
      (<c2> <e2>)
      ...
      (t <e>))
```

```
(cond (<c> S))
```

```
(cond (<c> S1)
      (t S2))
```

Dangling-Else Problem:

```
if c1
  if c2 S1
else S2
```

OR

```
if c1
  if c2 S1
  else S2
```

else associates with the closest unmatched if-

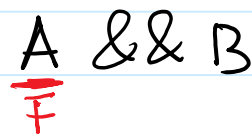
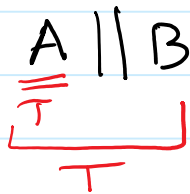
9 Control Flow - Selection

Thursday, October 15, 2020 8:35 AM

```
int x[N];  
int [] x = new int[N];  
if ( (i < N) && (x[i] > 0) ) {  
    }  
}
```

i = N
A - false

(short-circuited expr. evaluation)



Multi-way Selection

```
if month == 2:  
    if leapYear(year):  
        days = 29  
    else:  
        days = 28  
elif month == 1 or month == 3 or month == 5  
    or month == 7 or month == 8 or month == 10 or month == 12:  
    days = 31  
elif month in [4, 6, 9, 11]:  
    days = 30  
else:  
    // ERROR...
```

Switch/case

9 Control Flow - Multi-way Selection

Thursday, October 15, 2020 8:35 AM

C/C++/Java

switch (month) {

case 2: == }
break;

case 3:
case 5: } fall through
case 7:
case 8:
case 10: }

case 12: days = 31; break;

case 4:

case 6:

case 9:

case 11: days = 30; break;

default: // ERROR

}

string literals ok!

Ada

case month is

when 2 => ==

when 3|5|7|8|10|12 => days := 31;

when 4|6|9|11 => days = 30;

when others => ==

end case;

V6/14!