Introduction to Computers and Programming

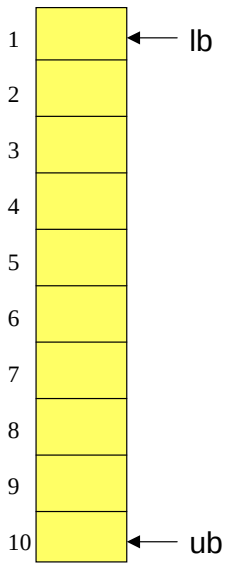
Prof. I. K. Lundqvist

Recitation 1
Mar 18 2004

Arrays

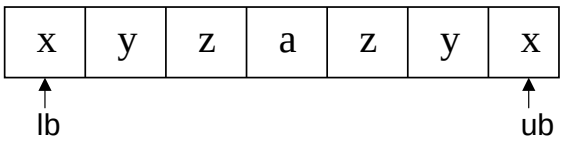
- Static structures of fixed size
  - lb: lower bound
  - ub : upper bound
  - index<lb | index>ub →constraint error

```
type My_Array is array (1 .. 10) of Integer;
                        lb     ub
```



Palindrome

- A palindrome is a string that reads the same both forward and backward



- Two questions:
  - How do you check equality?
  - When do you stop?

Evaluating Postfix

Read from left to right:

1. if a number is read, push it on the stack
2. if an operator is read, pop two numbers off the stack (the first number popped is the *second* binary operand)
3. apply the operation to the numbers, and push the result back onto the stack
4. when the expression is complete, the number on top of stack is the answer

5 3 + 4 + 1 +
5 3 + 10 *

Infix to Postfix

```
post_fix := ""
Create(Op_Stack)
for I in 1 .. Length loop
   If Is_Operand(expr(I)) = true then
         Append(post_fix, expr(I))

   If Is_Operator(expr(I)) = true then
         Process_Next_Operator(expr(I))
end loop
-- string post_fix has the result
```


Process_Next_Operator

```
Done := False
loop
    If Is_Empty(Op_Stack) or next_op is '(',
       push next_op onto Op_Stack
       set Done to True
    Elsif precedence(next_op) > precedence(top_operator)
       Push next_op onto Op_stack
       -- ensures higher precedence operators evaluated first
       Set Done to True
    Else
       Pop the operator_stack
       If operator popped is '('
             set Done to True
       Else
             append operator popped to post_fix string
    exit when Done = True
end loop
```

Infix to Postfix: Example

- Infix Expression

  3 + 5 * 6 – 7 * (8 + 5)

- Postfix Expression

  3 5 6 * + 7 8 5 + * –

Unary Operators

- '+' and '-' are symbols used for both binary and unary operations
- How do you distinguish between binary and unary operators?

Infix to Postfix: Example

- Infix Expression

  3 + 5 * – 6 – 7 * (8 + 5)

- Postfix Expression

  3 5 6 – * + 7 8 5 + * –