Vectors

Fundamentals of Computer Science

2010-2011

Ismael Etxeberria Agiriano

08/11/2010

Escuela Universitaria
de Ingeniería
Vitoria-Gasteiz

Ingeniaritzako
Unibertsitate Eskola
Vitoria-Gasteiz

eman ta zabal zazu

Universidad
del País Vasco

Euskal Herriko
Unibertsitatea

Index
Vectors

1. Introduction
2. Declaration
3. Access
4. Parameter passing
5. Complete access
6. Search

1. Introduction

- **Need**
  - Group data of the **same type** under a single name
  - Vectors, matrices, tables, arrays, lists, ...
- **Access**
  - We access the elements through an **index**
  - Operations over vectors are carried out element by element (i.e. a vector cannot be copied in a single operation)
- **Declaration**
  - They are declared specifying the **indexes**
  - We often don't know how many elements we shall need and we make provision
- **Parameter passing**
  - Vectors are always passed **by reference**

2. Declaration (I)

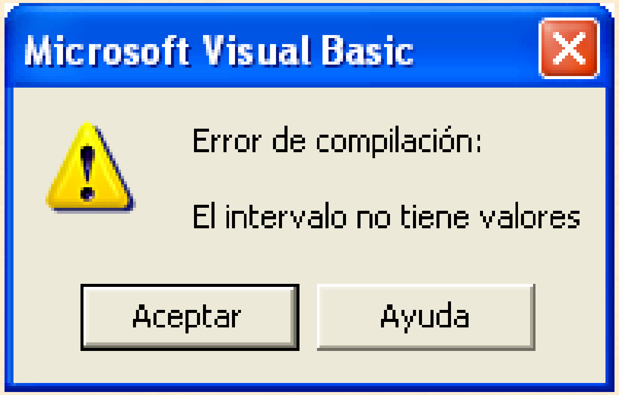
- **Specify the index interval**
  - The upper and lower indexes are specified
  - Example (**7 elements**):

```
Dim v(-1 To 5) As Double
```

-1	0	1	2	3	4	5
48.12	-256.901	18.001	-243.7	8.231	0.3818	4.21231

- **It must be a valid interval**
  - Example of error:

```
Dim v(10 To 1) As Integer
```



Microsoft Visual Basic

Error de compilación:

El intervalo no tiene valores

2.1 Declaration (II)

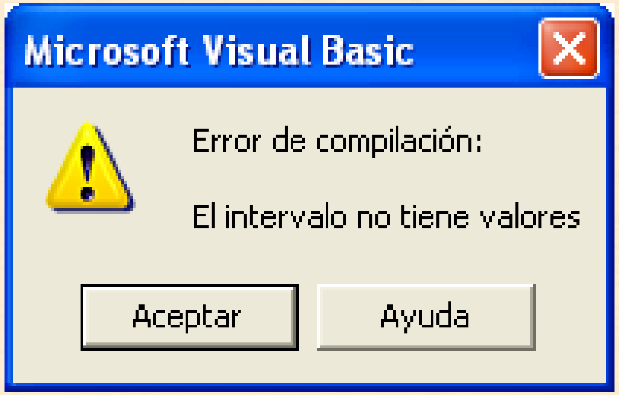
- **Specify the upper index (alternative)**
  - We may declare a vector specifying its upper index
  - The lower index is 0
  - Example (**7 elements**):

```
Dim ds(6) As String
```

0	1	2	3	4	5	6
"Mon"	"Tue"	"Wed"	"Thu"	"Fri"	"Sat"	"Sun"

- **It must be a valid interval**
  - Example of error:

```
Dim v(-1) As String
```


Microsoft Visual Basic

Error de compilación:

El intervalo no tiene valores

3. Access

- **Element**
  - We access an element indicating its index within parenthesis
  - Example:

```
pctRes.Print v(5)
```

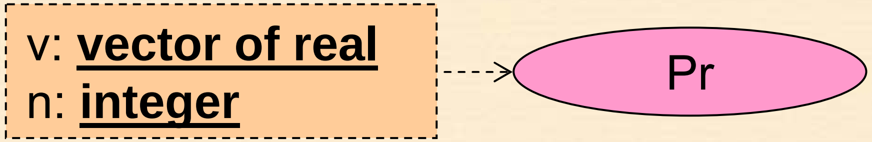
- **Extreme indexes of the vector**
  - Visual Basic allows knowing the extreme indexes of a vector:
    - **LBound**: Lower bound
    - **UBound**: Upper bound
  - We shall frequently not occupy the whole vector and the element in position **UBound** will have an undefined value (if we don't consider that VB initializes all values to 0)

4. Parameter passing (I)

- **By reference**
  - To pass a vector by value implies obtaining a copy of all its elements, which is often slow and unnecessary
  - The majority of the programming languages prevent passing vectors by value. If the programmer wants a copy, this must be carried out explicitly ("by hand") and then passed by reference
- **Flowchart**
  - To design the header of a subprogram we must pay attention on whether existing data are going to be read from the vector (input parameter) and/or data are going to be modified from the vector (output parameter)

4. Parameter passing (II)

- **Visual Basic**
  - The VB header does not clarify which vector parameter is input and/or output as we always pass them by reference
  - We specify that a parameter is a vector by opening and closing parenthesis but without specifying size or indexes
  - We may call a subprogram with vectors of different sizes
  - Example of the header of a subprogram that receives a vector (e.g. to write its contents on a picture box):

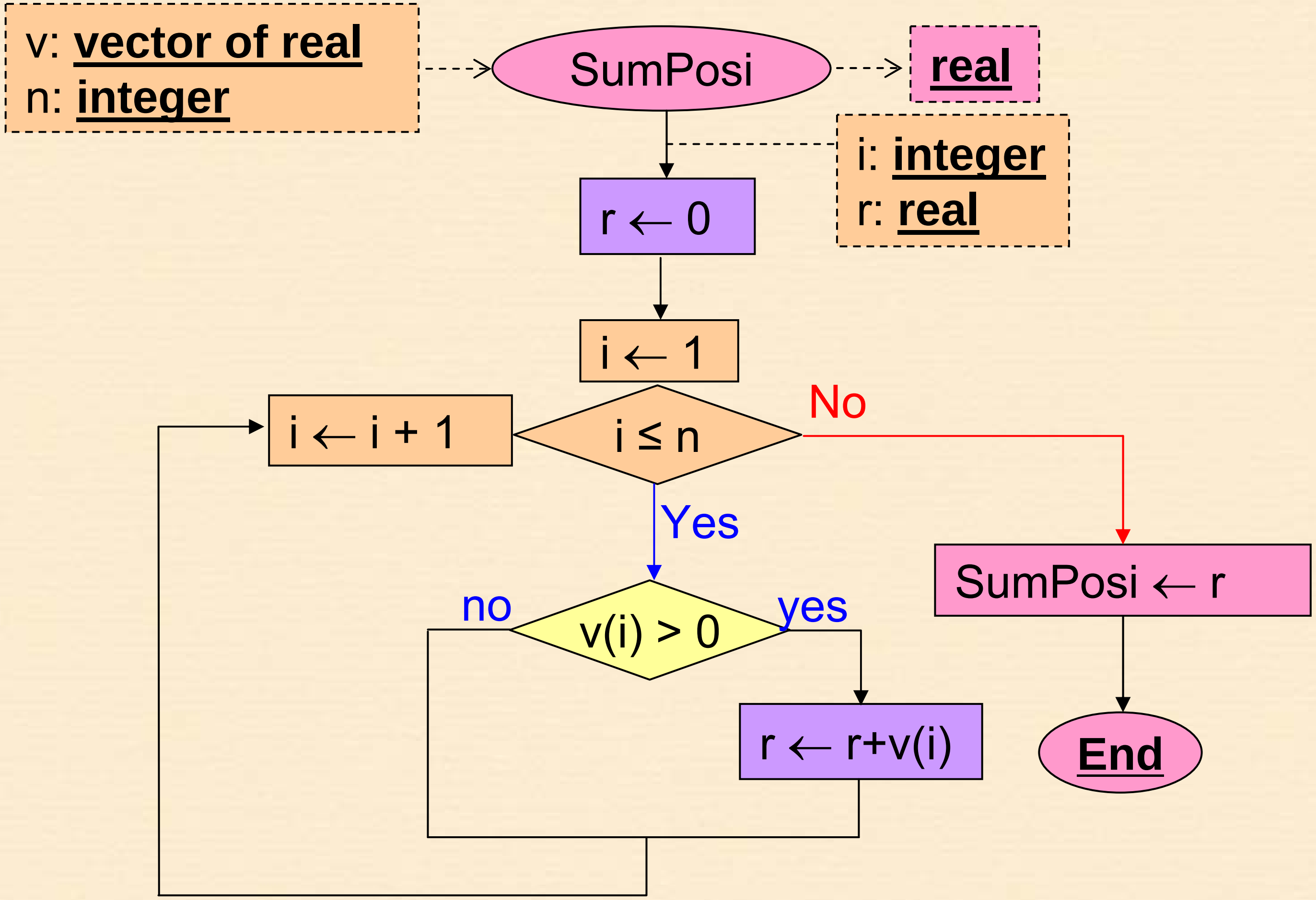


```
Sub Pr (ByRef tbl() As Double, ByVal n As Integer)
```

5. Complete access

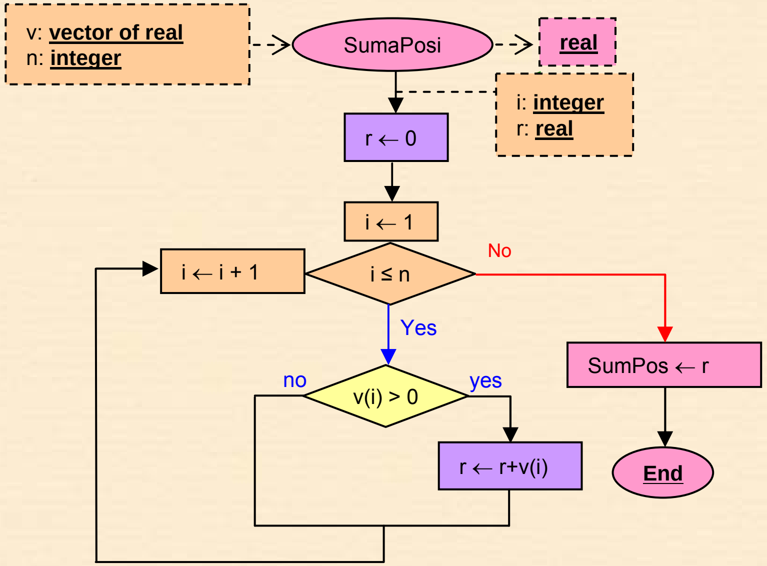
- **Examples**
  - Sum all the elements of a vector
  - Count the number of elements in a vector that fulfil a property
  - Carry out an operation over all the elements of a vector
- **General resolution model**
  - We utilize a `For` loop to access the complete vector
- **Visual Basic**
  - From `1` to `n` or from `0` to `n-1` or from `0` to `n`
  - From `ini=Lbound(v)` to `ini + n - 1`

5.1 Sum positive numbers



5.2 Sum positive numbers

```
Function SumPosi (ByRef v() As Double, ByVal n As Integer) _
                  As Double
  Dim i As String
  Dim r As Double
  r = 0
  For i = 1 To n Step 1
    If v(i) > 0 Then
      r = r + v(i)
    End If
  Next i
  SumPosi = r
End Sub
```



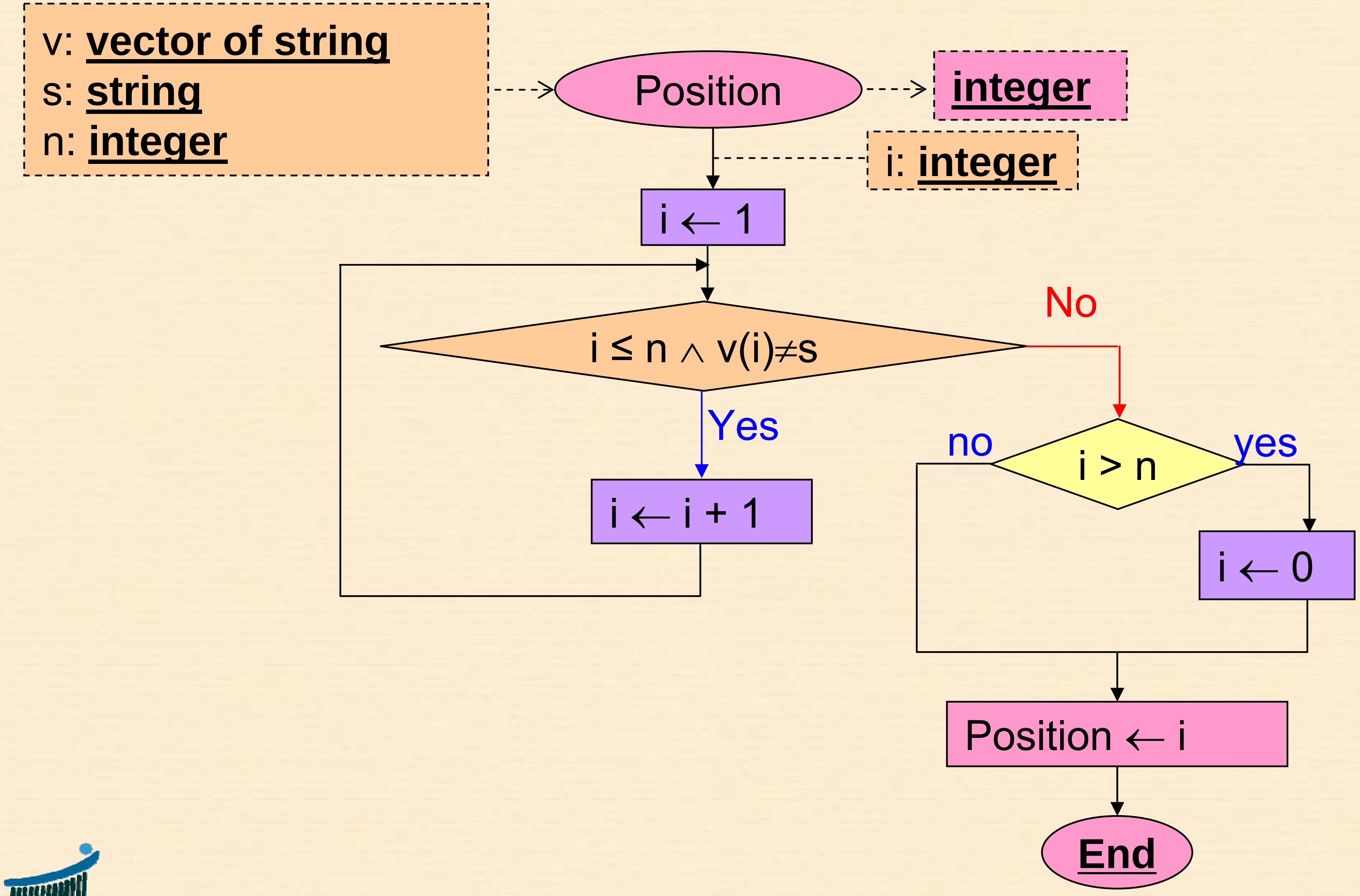
6. Searching

- **Examples**
  - Search for the position of a number or a string in a vector
  - Search for the first elements that fulfils a property
- **General resolution model**
  - There are two stopping conditions:
    1. We have found what we were looking for
    2. We run out all opportunities (tried all elements)
  - Convert the `For` loop into a `While` loop to access all elements or stop if an appropriate candidate is found
- **Visual Basic**
  - From `1` to `n And Not Found`
  - From `ini=LBound(v)` to `ini + n`

Escuela Universitaria de Ingeniería Vitoria-Gasteiz Ingeniaritzako Unibertsitate Eskola Vitoria-Gasteiz

eman ta zabal zazu

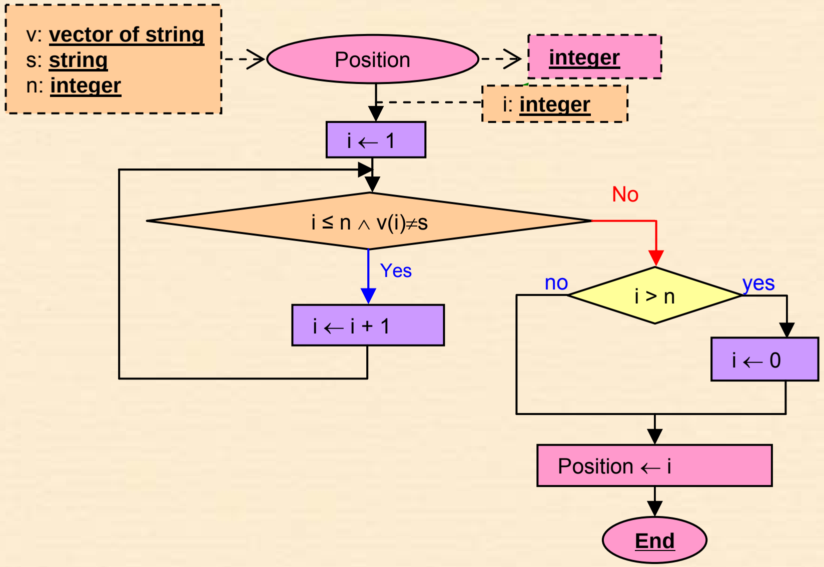
6.1 Search for a name in a list



6.2 Search for a name in a list

```
Function Position (ByRef v() As String, ByVal s As String, _
                   ByVal n As Integer) As Integer
  Dim i As String

  i = 1
  While i <= n And v(i) <> s
    i = i + 1
  Wend
  If i > n Then
    i = 0
  End If
  Position = i
End Sub
```



Escuela Universitaria
de Ingeniería
Vitoria-Gasteiz

Ingeniaritzako
Unibertsitate Eskola
Vitoria-Gasteiz

eman ta zabal zazu

Universidad
del País Vasco

Euskal Herriko
Unibertsitatea