

Lecture 4

2.2 Using Objects: the `list` class

4.4 Conditional Statements (review)

2.2 Using Objects: the `list` class

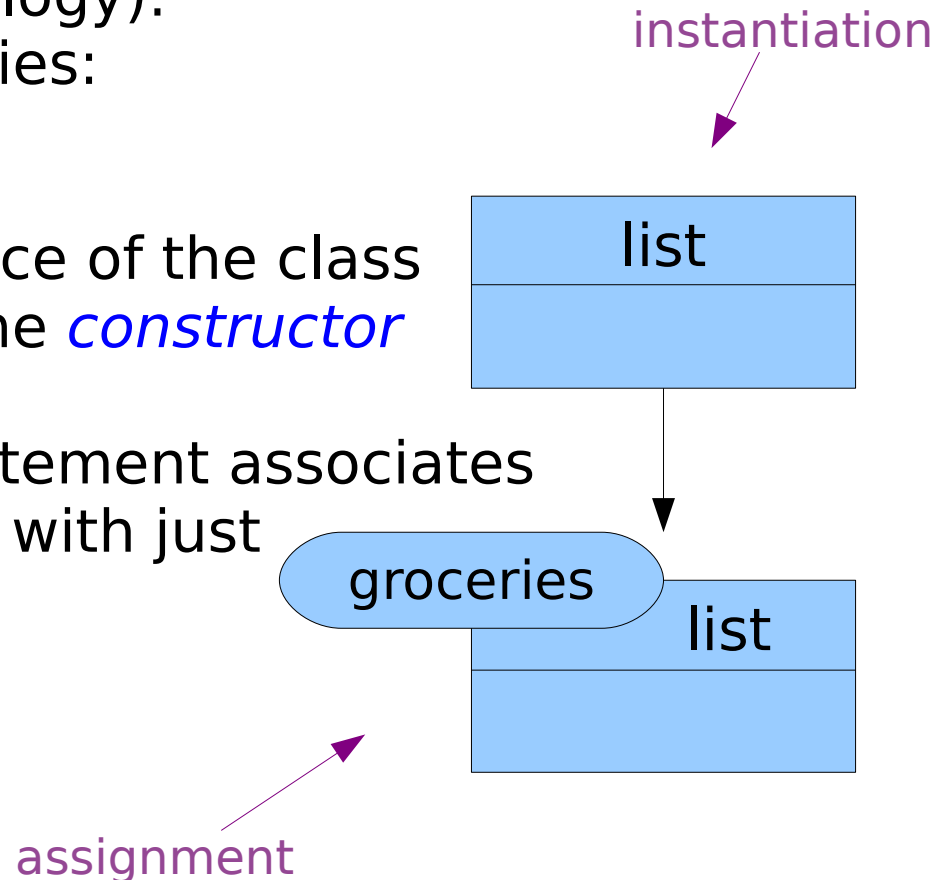
- a built-in class
- used to maintain an ordered list of items

Example(REVIEW of terminology):
Let's make up a list of groceries:

```
groceries = list()
```

- `list()` creates an instance of the class `list` (by invoking the *constructor* of the class)

- then the assignment statement associates the identifier `groceries` with just created object



2.2 Using Objects: the `list` class

```
def main():  
    groceries = list()  
  
    groceries.append('milk')  
    groceries.append("bread")  
    groceries.append("cream cheese")  
  
    groceries.insert(0, "sour cream")  
  
    groceries.remove("bread")  
  
    groceries.sort()  
  
    print len(groceries)
```

main()

print the current
length of the list

adding elements to
the end of the list

adding 'sour cream'
to the top of the list

removing the first
occurrence of
'bread' from the list

sort list in
alphabetical order

See more methods (functions) on page 42

*See an example of a program in
list-example1.py*

4.4 Conditional Statements

Syntax of the *if-statement*:

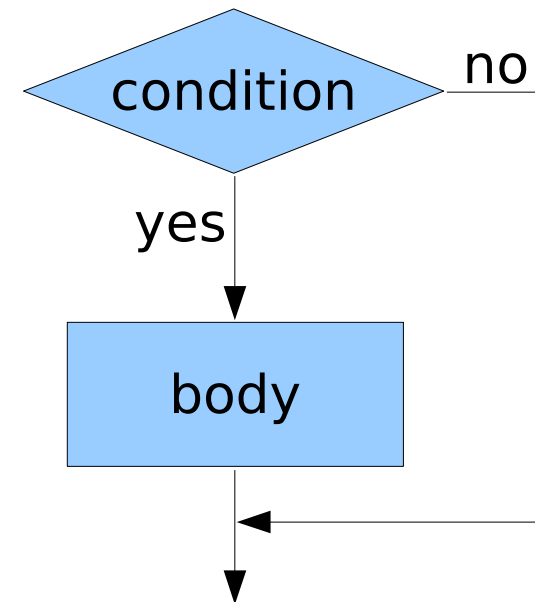
```
if <condition>:  
  body
```

example: *condition*

```
if t>90:  
  print "Heat Warning!"
```

body

activity diagram
(flowchart) of a simple
if-statement:



4.4 Conditional Statements

Syntax of the *if-else statement*:

```
if <condition>:  
    body1  
else:  
    body2
```

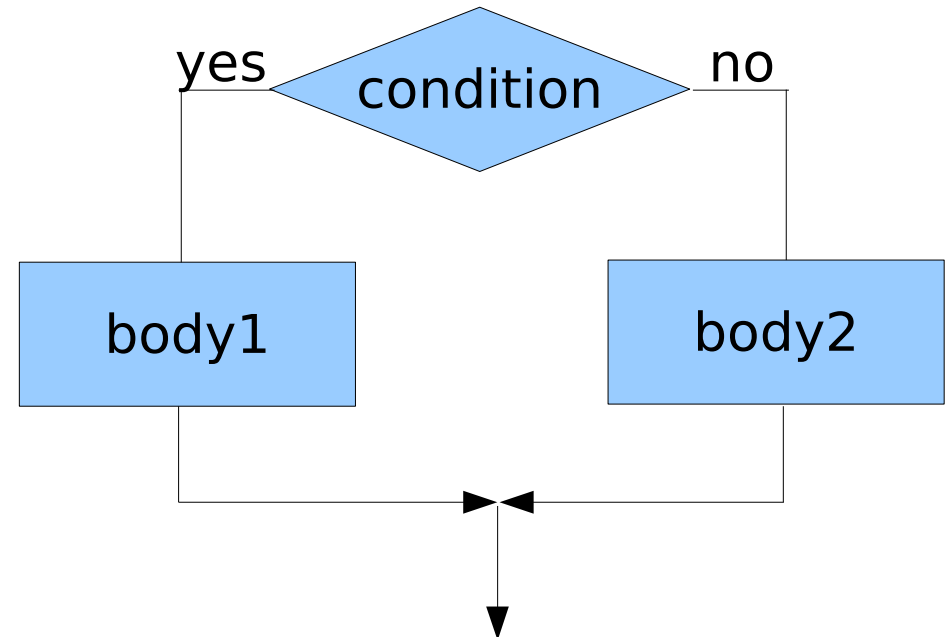
example: *condition*

```
if t>90:  
    print "Heat Warning!"  
else :  
    print "everything is fine"
```

body1

body2

activity diagram
(flowchart) of an
if-else statement:



4.4 Conditional Statements

Syntax of the *if-elif statement*:

```

if <condition>:
    body1
elif <condition2>:
    body2
...
elif <conditionN>:
    bodyN
else:
    bodyN1
  
```

Same as to the right,
but more compact
and elegant →

```

if <condition>:
    body1
else:
    if <condition2>:
        Body2
    else:
        if <condition3>:
            Body3
        else: . . .
            . . .
                else:
                    if <conditionN>:
                        bodyN
                    else:
                        bodyN1
  
```

Example: Assigning a grade for a test

<60 : F

60-62: D-

63-66: D

67-69: D+

70-72: C-

73-76: C

77-79: C+

80-82: B-

83-86: B

87-89: B+

90-92: A-

93-96: A

97-100: A+

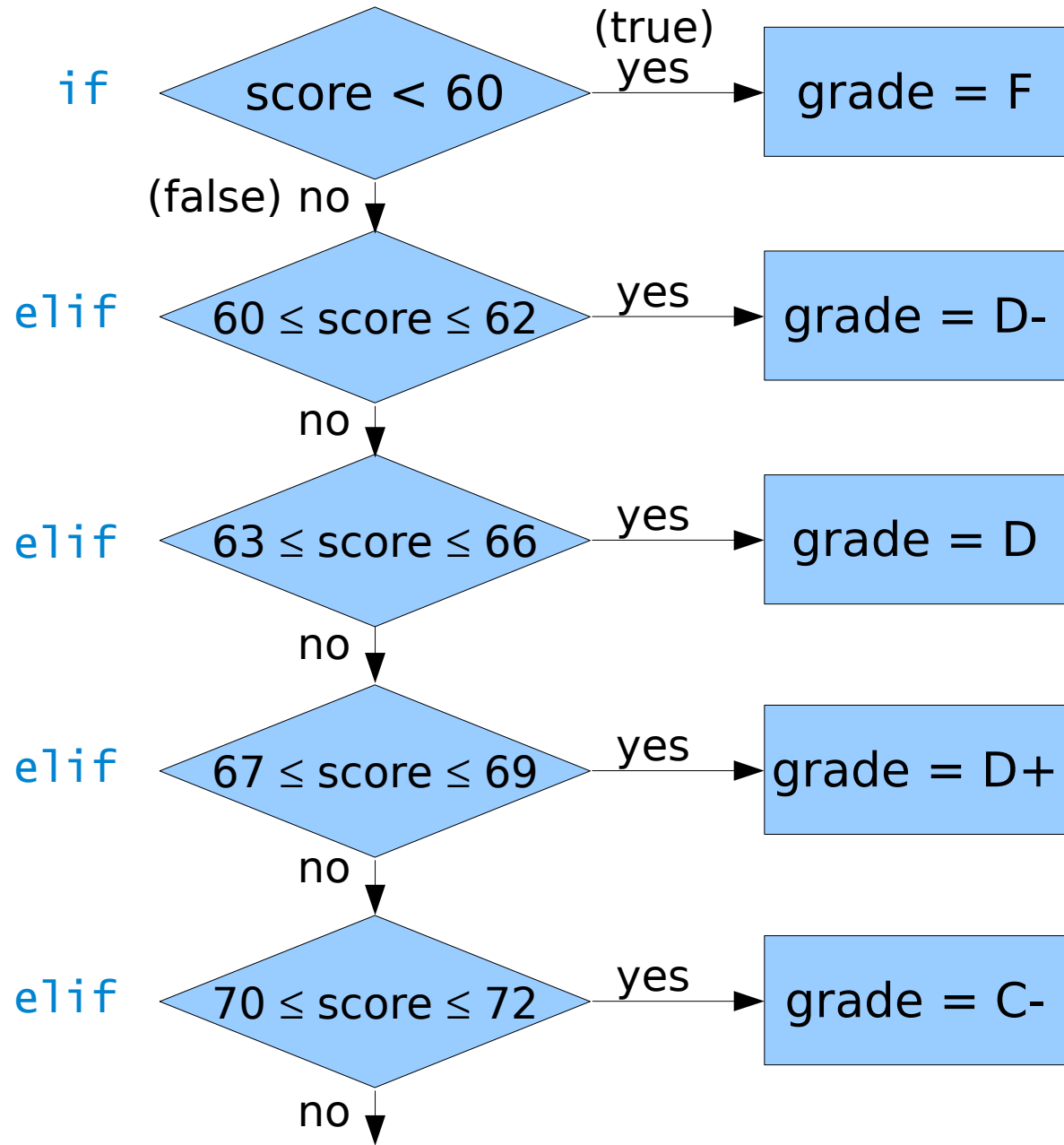
Let's write a program that is based on the test score assigns the test grade.

Analysis:

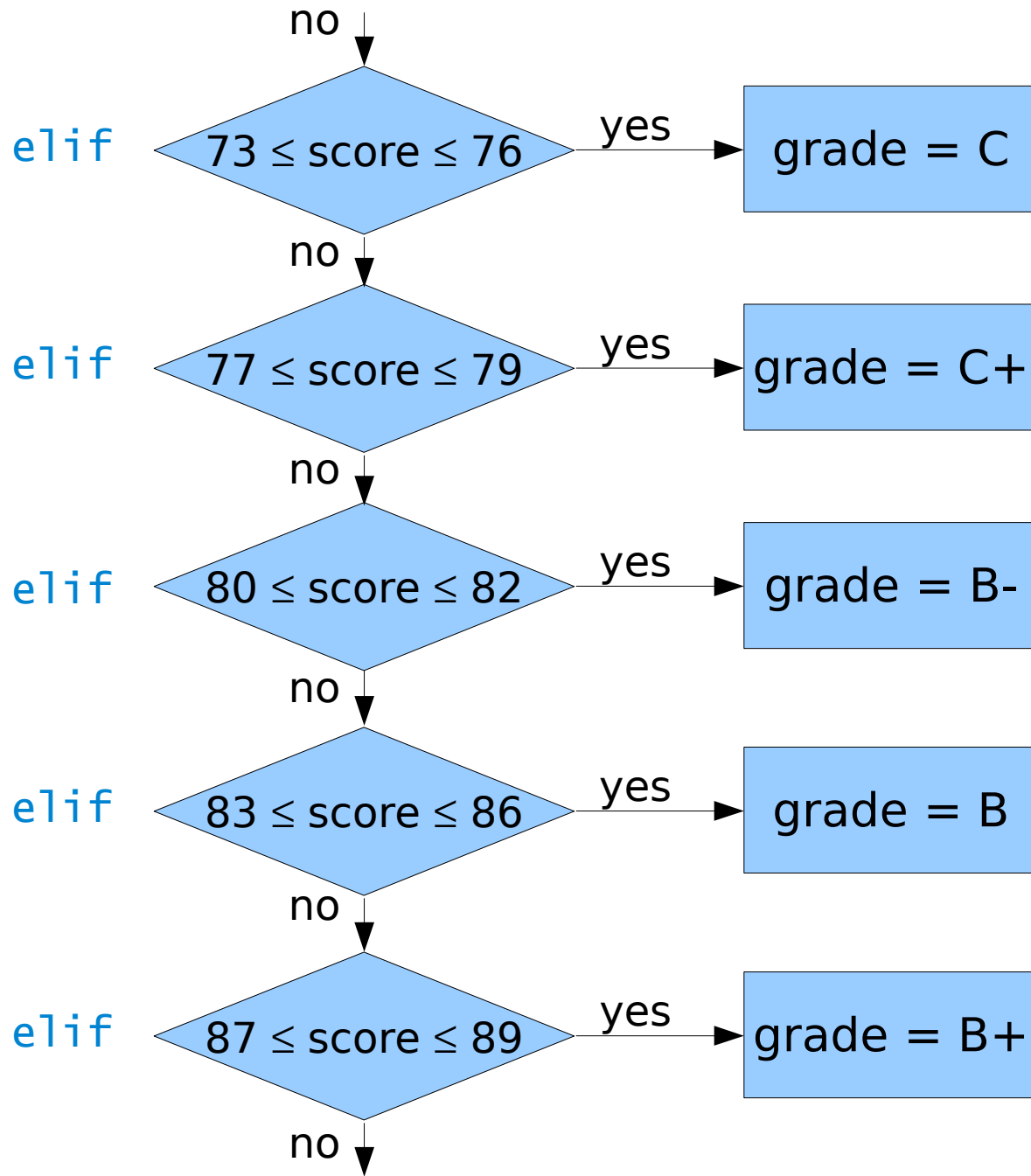
Input: a whole number less than 101 (which is the test score)

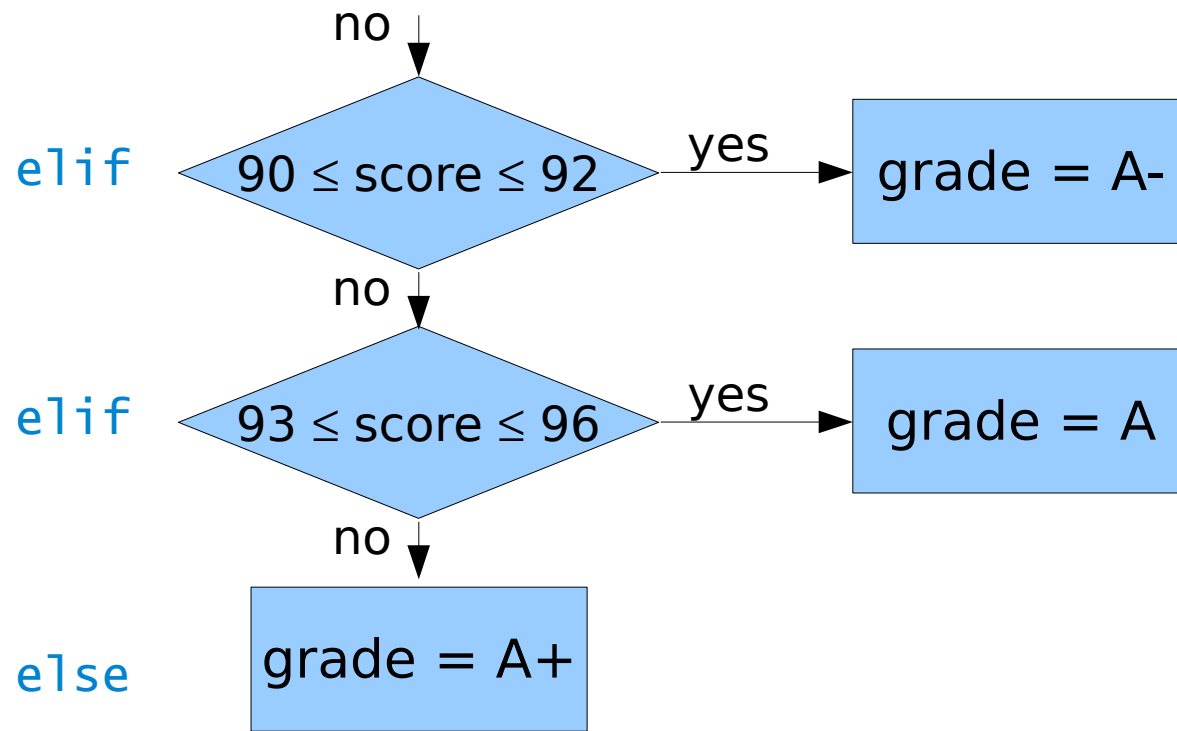
Output: grade that corresponds to the test score

Activity diagram:



At the **Design stage** we need to present the activity diagram (flowchart) only





Then goes the **implementation** (programming) **stage**.

See `scoreToGrade.py`

Homework Assignment

HW3:

Write a program that takes a list of strings as an input (from user, from keyboard), echoes it to the display and then allows the user to play with it, i.e. each time user is given the result along with the menu of things he/she can do with it.

snapshot:

Here is what we have at this point:

```
['Hello', 'buy', 'mind', '5', 'abc', '12', '0', '1243', 'thank you',  
'sorry', '432', 'mind', '5', 'Hello', 'George']
```

What would you like to do?

1. Find a first occurrence of an item
2. Find all occurrences of an item
3. Add an item to the end of the list
4. Remove an item from the list
5. Insert an item into specified position
6. Reverse the order of the elements in the list
7. Replace the element at the specified position with something else
8. Quit the program

next
slide



When submitting HW3:

Provide the analysis and the design of your project (see example we did in class).

The implementation (program) itself must be well-commented.

Also make sure that you provide a little bit of description of the program to the user.

You will be given two separate grades:

One for the program and one for the provided documentation (analysis, design, comments, description of the program's work)