How to THINK like a Programmer

Problem Solving for the Bewildered

paul vickers

Problem Solving 5

More on Choices

John Brosnan,
Computing Dept, ITT

Aim

- In this lesson we examine more decision making
  - The `if-else` structure for 2 way decisions
  - Multi-way selections with nested `else-if`
  - Logical and relational operators
  - Nested selections

ACTIVITY

Write an `if-else` construct that chooses between wearing sandals or shoes depending on whether it is raining

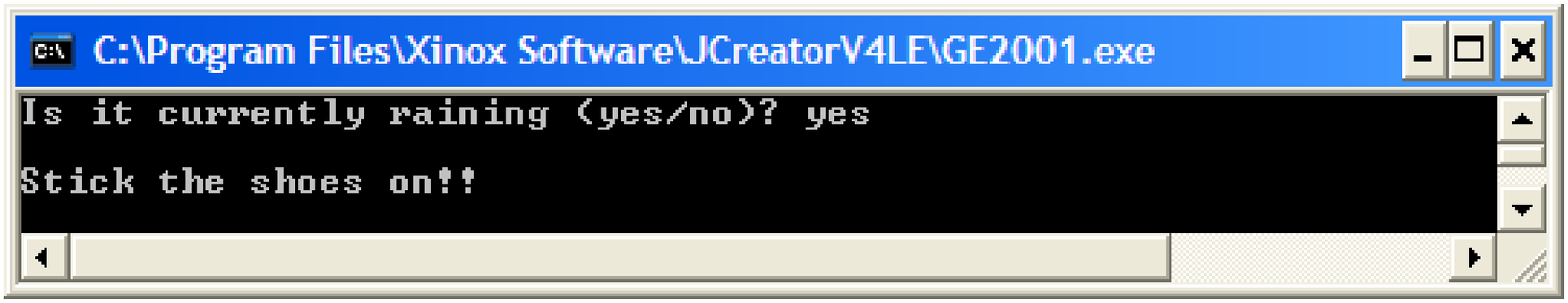
Solution

Pseudocode solution to determine whether to wear sandals or shoes:

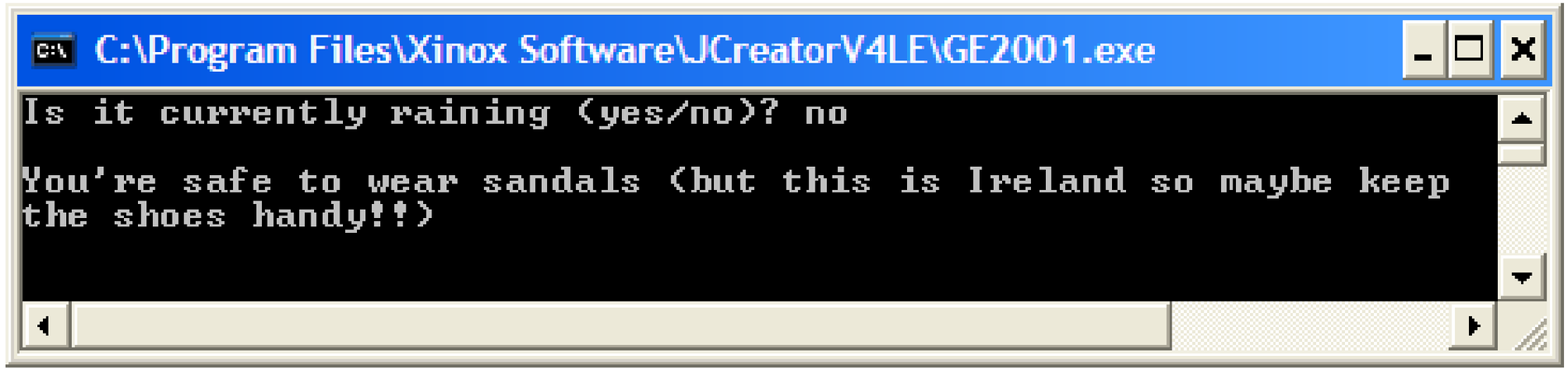
```
if (it is raining)
    wear shoes
else
    wear sandals
endif
```

Now you should try to write a Java program based on the pseudocode solution. Your program should run according to the following sample screenshots

Run 1: In this case it is raining:



Run 2: In this case it is dry:



ACTIVITY

Stocksfield High School gives email addresses to all its pupils in the form:
firstName.surnameName@stocksfieldhigh.ac.uk

Teachers are given email addresses of the form:
firstInitial.secondInitial.familyName@stocksfieldhigh.ac.uk

where the second initial may or may not not be present.

The table below shows some example teacher and pupil names and the corresponding email addresses.

Teachers	***email***	***Pupils***	***email***
Henry Higgins	*h.higgins@...*	*Emily Harris*	*emily.harris@...*
Alfred P. Doolittle	*a.p.doolittle@...*	*Sarah Jane Smith*	*sarah.smith@...*
Jennifer P.D. Quick	*j.p.quick@...*	*John Dobby*	*john.dobby@...*

Design an algorithm that reads in a user-supplied single email address and then displays the surname of the owner of that email address and whether they are a pupil or a teacher.

For example, if the user typed in
emily.harris@stocksfieldhigh.ac.uk

the algorithm would display

Harris:Pupil

If the user typed
a.p.doolittle@stocksfieldhigh.ac.uk

The algorithm would display
Doolittle:Teacher

Note that the algorithm should be able to handle any valid Stocksfield High email address, not just the ones listed in the table.

Solution

Pseudocode solution to determine the surname corresponding to an email address and whether it belongs to a student or a teacher:

```
Prompt for and read in the email address
// next up is the surname part
Display characters between '@' and the preceding '.'
if (length of name before the '.' > 1 character)
// must be dealing with a pupil in that case
        Display the text ':Pupil'
else
// must be dealing with staff in that case
        Display the text ':Teacher'
endif
```

Note that you are not expected to be able to code a solution to the above problem yet because all solutions to it require either the use of loops and/or the use of some special Java methods that we have yet to encounter. If you figured out the pseudocode solution, it is a very good sign that your problem-solving abilities are in good nick.

Multi-way selections

- What if we need more than just an either/or? What if we have multiple conditional actions? Consider the scenario:
  - Parcels 5 kg and under should be labelled ‘light’, over 5 kg but lighter than 10 kg is ‘medium’, and 10 kg or over is ‘heavy’.

    We could do this:

```
if (parcel weight up to and including 5 kilos)
   Add 'light' sticker
endif
if (parcel weight more than 5 and less than 10 kilos)
   Add 'medium' sticker
endif
if (parcel weight 10 kilos or over)
   Add 'heavy' sticker
endif
```

- But this solution is inefficient – we can extend if-else

The Nested `else-if` construct

- Now our pseudocode becomes:

```
if (parcel weight up to and including 5 kilos)
   Add 'light' sticker
else if(parcel weight  more than 5 and  less than 10kilos)
   Add 'medium' sticker
else
   Add 'heavy' sticker
endif
```

- Notice how each condition makes use of the results of the previous one, and the last `else` has no condition

Logical Operators

- Note the use of the word “and” in the expression

```
parcel weight more than 5 and less than 10 kilos
```

- It is called a logical operator and is used to join 2 conditions.

  With “and”, both conditions must be true if the associated action block is to be executed.

- The 2 other logical operators in pseudocode are “or” and “not”.
- With “or”, as long as one of the conditions is true, the associated action block will execute.
- With “not”, as long as the condition is false, the associated action block will execute since “not false” is actually true.

Relational Operators

- Writing things like `parcel weight up to 5 kilos` is quite verbose
- We can use the **relational operators** to make it neater

Operator	pseudocode
Less than	<
Less than or equal to	≤
Equals (equality)	=
Greater than or equal to	≥
Greater than	>
Not equal to (inequality)	≠

The Nested `else-if` construct with relational operators

- Now our pseudocode becomes:

```
if (parcel weight ≤ 5 kilos)
   Add 'light' sticker
else if(parcel weight  < 10 kilos)
   Add 'medium' sticker
else
   Add 'heavy' sticker
endif
```

- Notice how

```
parcel weight more than 5 and less than 10 kilos
```

has become `parcel weight < 10 kilos`. This is because if the `else if` section is reached, the weight must have been over 5kg

ACTIVITY

What's wrong with the following pseudocode to assign grades?

```
if (mark ≥ 80)
   set grade to 'A'
else if (mark ≥ 70) and (mark ≤ 79)
   set grade to 'B'
else if (mark ≥ 60) and (mark ≤ 69)
   set grade to 'C'
else if (mark ≥ 50) and (mark ≤ 59)
   set grade to 'D'
else if (mark ≥ 40) and (mark ≤ 49)
   set grade to 'E'
else if (mark < 40)
   set grade to 'F'
endif
```

Solution

- The structure does not make use of previous conditions and so is very inefficient, carrying out unnecessary tests for conditions that are already known. It should be:

```
if (mark ≥ 80)
   set grade to ‘A’
else if (mark ≥ 70)
   set grade to ‘B’
else if (mark ≥ 60)
   set grade to ‘C’
else if (mark ≥ 50)
   set grade to ‘D’
else if (mark ≥ 40)
   set grade to ‘E’
else
   set grade to ‘F’
endif
```

Nested selections

- Consider the following algorithm for deciding the number of days in a month

```
if (month = 4) or (month = 6) or (month = 9) or (month = 11)
    set days in month to 30
else if (month = 2)
    if  (this is a leap year)
        set days in month to 29
    else
        set days in month to 28
    endif
else
    set days in month to 31
endif
```

this selection is nested within the `else-if` part

ACTIVITY

A school has just changed its grading system. An A grade is given for marks between 4 and 5, a B for marks of at least 3.5, a C for marks of at least 3.0, a new pass grade D for marks of at least 2.0 and an F for every mark lower than 2.0

Write the pseudocode solution for this situation

Solution

Pseudocode for assigning grades:

Prompt for and read in a mark

if (mark ≥ 4)

    set grade to ‘A’

else if (mark ≥ 3.5)

    set grade to ‘B’

else if (mark ≥ 3)

    set grade to ‘C’

else if (mark ≥ 2)

    set grade to ‘D’

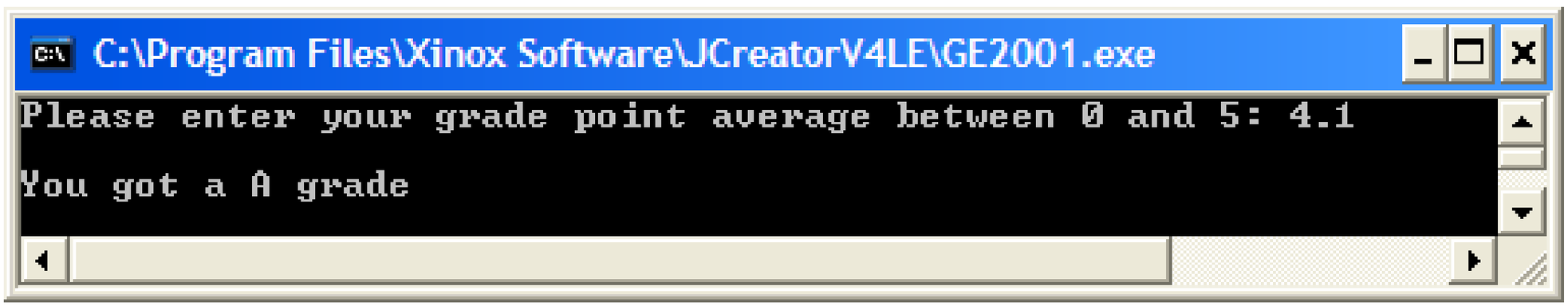
else

    set grade to ‘F’

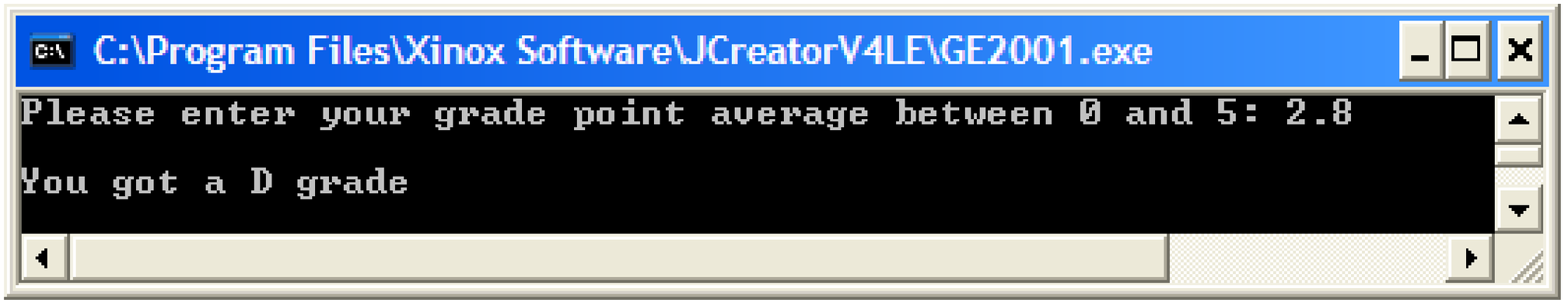
endif

Now you should try to write a Java program based on the pseudocode solution. Your program should run according to the following sample screenshots (only 2 scenarios are shown here for brevity but make sure you test your program so it works for each range)

Run 1: In this case the GPA is 4.1



Run 2: In this case the GPA is 2.8



ACTIVITY

Honest Brian's Insurance sells car insurance policies. The company is owned by Brian who is a cautious type and so hikes the premiums for young male drivers (who are statistically much more likely to make a claim). The basic premium charge is 3% of the value of the vehicle. This figure is raised by 11% for male drivers under 25 years of age, and by 6% for female drivers under the age of 21. A further €250 is added to the premium of any driver who has had any kind of speeding ticket.

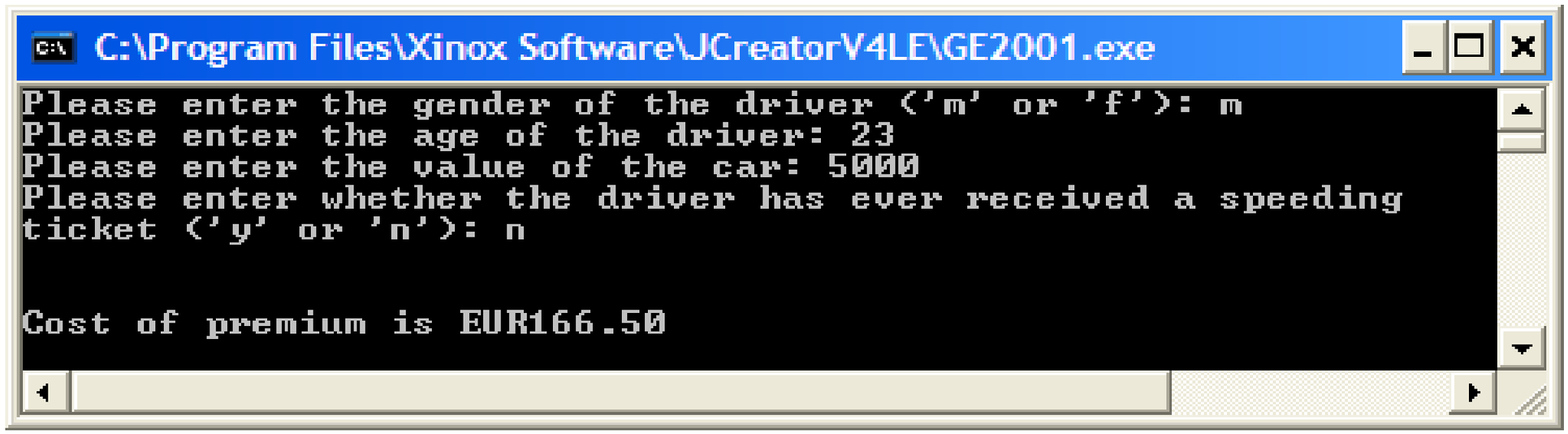
Design an algorithm to calculate the insurance premiums for drivers in this case.

Pseudocode for determining the insurance premium for a driver:

1. Prompt for and read in the gender of the driver
2. Prompt for and read in the age of the driver
3. Prompt for and read in the value of the car
4. Prompt for and read in whether the driver has had a speeding ticket
5. set the premium to car value x 0.03
6. if (gender = ‘m’) and (age < 25)
   6.1 set premium to premium x 1.11
7. else if(gender = ‘f’) and (age < 21)
   7.1 set premium to premium x 1.06
   endif
8. if (driver has speeding ticket)
   8.1 set premium to premium + 250
   endif

Now you should try to write a Java program based on the pseudocode solution. Your program should run according to the following sample screenshots (only 2 scenarios are shown here for brevity but make sure you test it fully):

Run 1 : Here we have a male driver under 25 with a EUR5000 car who never received a speeding ticket



Run 2 : Here we have a female driver over 21 with a EUR10000 car who did receive a speeding ticket