- Previous class:
  - Intro to computer programming
  - Variables & assignment
  - Input & output
  - Script
  - Calling functions for graphics
- Now:
  - Branching

1

A script (program) is a file with a sequence of commands

```
% Quadratic equation solver
a= input('Enter a: ');
b= input('Enter b: ');
c= input('Enter c: ');
d= b^2 -4*a*c;
r1= (-b-sqrt(d))/(2*a)
r2= (-b+sqrt(d))/(2*a)
```

Comment begins with %

Variable holds a value

Semi-colon suppresses "echo"

Assignment operator: value on RHS is assigned to variable named on LHS

A file with the extension `.m`

2

```
% Quadratic equation solver
a= input('Enter a: ');
b= input('Enter b: ');
c= input('Enter c: ');
d= b^2 -4*a*c;
if d >= 0
   r1= (-b-sqrt(d))/(2*a)
   r2= (-b+sqrt(d))/(2*a)
else
   fprintf('Complex roots\n')
end
```

File: `qSolver.m`

Another version of the program…

```
% Quadratic equation solver
a= input('Enter a: ');
b= input('Enter b: ');
c= input('Enter c: ');
d= b^2 -4*a*c;
% Write your if-statement below...
```

The `if-else` construct

`if` <*condition*>

  statements to execute if condition is true

`else`

  statements to execute if condition is false

`end`

Only one block of statements will be executed!

6

Relational operators

`<`	Less than
`>`	Greater than
`<=`	Less than or equal to
`>=`	Greater than or equal to
`==`	Equal to
`~=`	Not equal to

7

Suppose I don't care about the values of the roots—I just want to know if the roots are comlex.

```
% Quadratic equation solver
a= input('Enter a: ');
b= input('Enter b: ');
c= input('Enter c: ');
d= b^2 -4*a*c;
if d < 0
   fprintf('Complex roots\n')
end
```

The if construct

```
if  <condition>

    statements to execute if condition is true

end
```

10

Logical AND

Q. When is a real number x in the interval [L,R]?

A. If x is greater than or equal to L and less than or equal to R.

```
if (x>=L && x<=R)
  fprintf('x is in [L,R]')
else
  fprintf('x is not in [L,R]')
end
```

11

Logical OR

Q. When is a real number x not in the interval [L,R]?

A. If x is less than L or less greater than R.

```
if (x<L || x>R)
  fprintf('x is not in [L,R]')
else
  fprintf('x is in [L,R]')
end
```

12

Boolean expressions

- They involve comparisons.
- They have a value that can be thought of as either true or false.

Example:

1. Variables a, b, and c have positive real values. Can we make a triangle with sides that have those values? Yes if the following is true:

13

2. Variable x has a positive integer value. Is it divisible by 3 and 5? Yes if the following is true:

15

3. Variable y has a positive integer value. Does it name a non-leap year? Yes if the following is true:

Hint: Y is an “ordinary” year if it is not divisible by 4 or if it is a century year not divisible by 400.

17

“false” is 0, “true” is non-zero

X, Y represent boolean expressions.
E.g., d>3.14

X	Y	X && Y “and”	X || Y “or”	~X “not”
1	1			
1	0			
0	1			
0	0			

23

Always use logical operators for multiple conditions

Why is it wrong to use the expression

`L <= x <= R`

for checking if *x* is in [*L*,*R*]?

Example: Suppose **`L`** is 5, **`R`** is 8, and **`xc`** is 10. We know that 10 is not in [5,8], but the expression **`L <= xc <= R`** gives...

24

```
% Find number of days in month m
m= input('Which month? ');




fprintf('Month %d has %d days\n',...
        m, days);
```

```
% Find number of days in month m
m= input('Which month? ');
```

Fill in the necessary code.

There are 3 possibilities: 30, 31, or 28 days.
So we need to choose 1 among 3 options.

```
fprintf('Month %d has %d days\n',...
        m, days);
```

28

The `if-elseif-else` construct

```
if  <condition 1>
   statements to execute if condition 1 is true
elseif  <condition 2>
   statements to execute if condition 2 is true
elseif  <condition 3>
   statements to execute if condition 3 is true
else
   statements to execute if condition 3 is false
end
```

Use this construct when there are many alternatives. Only one block of statements will be executed.

31

Things to know about the if construct

- ____________ branch of statements is executed
- There can be ____________ **elseif** clauses
- There can be ____________ **else** clause
- The **else** clause ________________ in the construct
- The **else** clause ____________________ (boolean expression)

32

Does this program work?

```
score= input('Enter score: ');
if score>55
     disp('D')
elseif score>65
     disp('C')
elseif score>80
     disp('B')
elseif score>93
     disp('A')
else
     disp('Not good…')
end
```

yes

no

34

```
% Find number of days in month m
m= input('Which month? ');
if m==2
   days= 28;

elseif rem(m,2)==1 && m<=7 || ...
       rem(m,2)==0 && m>=8
   days= 31;
else
   days= 30;
end
fprintf('Month %d has %d days\n',...
        m, days);
```

Let's re-organize …

35

```
% Find number of days in month m
m= input('Which month? ');
if m==2
   days= 28;
else
   if rem(m,2)==1 && m<=7 ||...
      rem(m,2)==0 && m>=8
      days= 31;
   else
      days= 30;
   end
end
fprintf('Month %d has %d days\n',...
        m, days);
```

Nested if statements

```
% Find number of days in month m
m= input('Which month? ');
if m==2
   days= 28;
else   % All months other than Feb



end
fprintf('Month %d has %d days\n',...
        m, days);
```

```
% Find number of days in month m
m= input('Which month? ');
if m==2
   days= 28;
else   % All months other than Feb
   if rem(m,2)==1 && m<=7 ||...
      rem(m,2)==0 && m>=8
      days= 31;
   else
      days= 30;
   end
end
fprintf('Month %d has %d days\n',...
        m, days);
```