Construct Project Flowchart

Dr. Mark E. Lehr

Author: Dr. Mark E. Lehr
Created on July 7th, 2017, 11:32 AM
Purpose: Flowchart Exercise of each construct

System Libraries
I/O Objects
Standard Namespace

User Libraries
none

Global Constants
none

Function Prototypes
none

main

Declare variables
x
msg

Prompt for Input
Input the value of x, x must be integer >= 0 and <= 2"
x

page 2
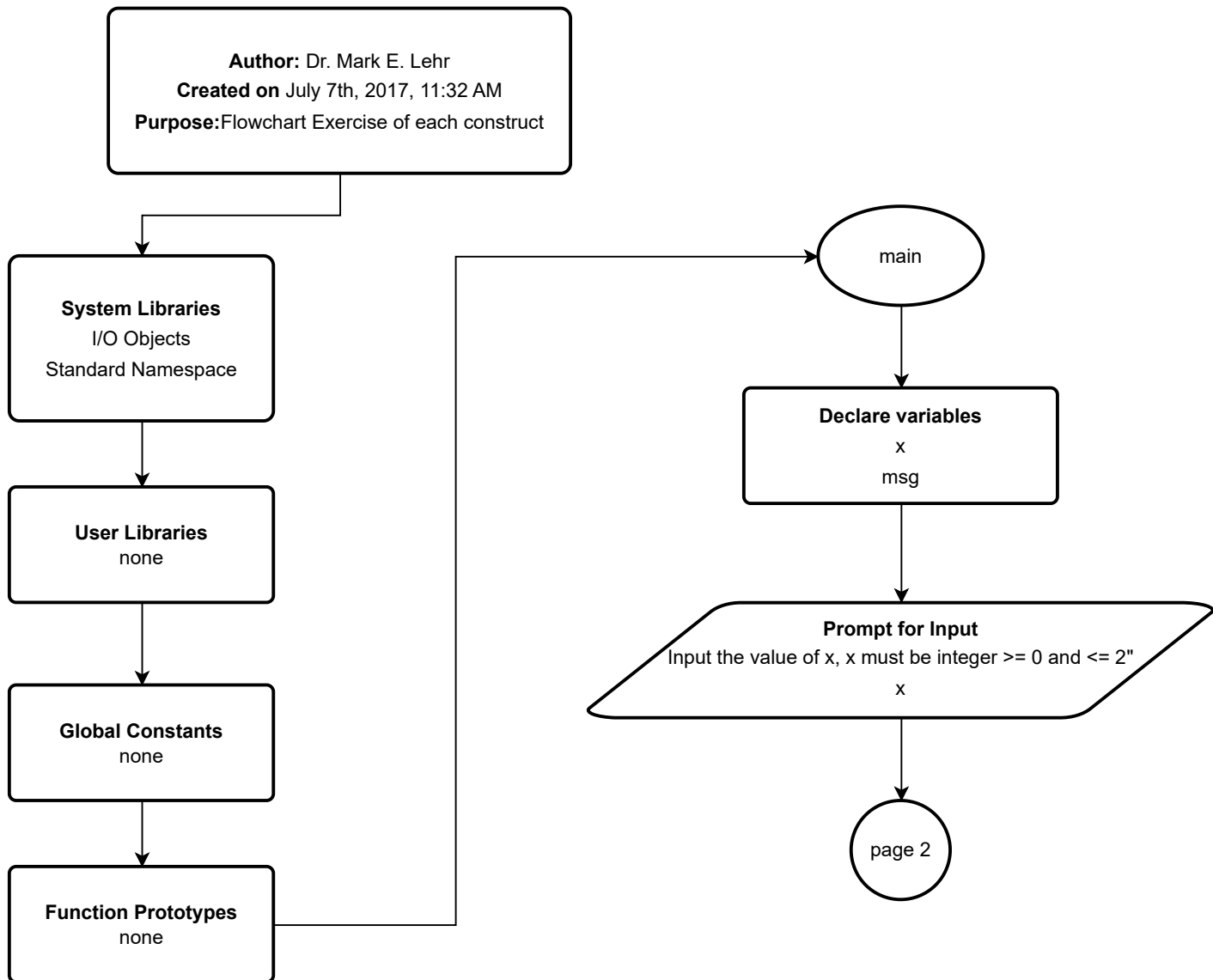
3 Independent If's and Single Dependent if-else if-else

page 2

Independent if's

Dependent if-else if-else

```
x<0 ──True──> Output "Value too Small"
  │
False
  │
  v
x>2 ──True──> Output "Value too Large"
  │
False
  │
  v
x>=0 && x<=2 ──True──> Output "In Range [0,2]"
  │
False
```

```
x<0 ──True──> Output "Value too Small"
  │
False
  │
  v
x>2 ──True──> Output "Value too Large"
  │
False
  │
  v
Output "In Range [0,2]"
```

page 3
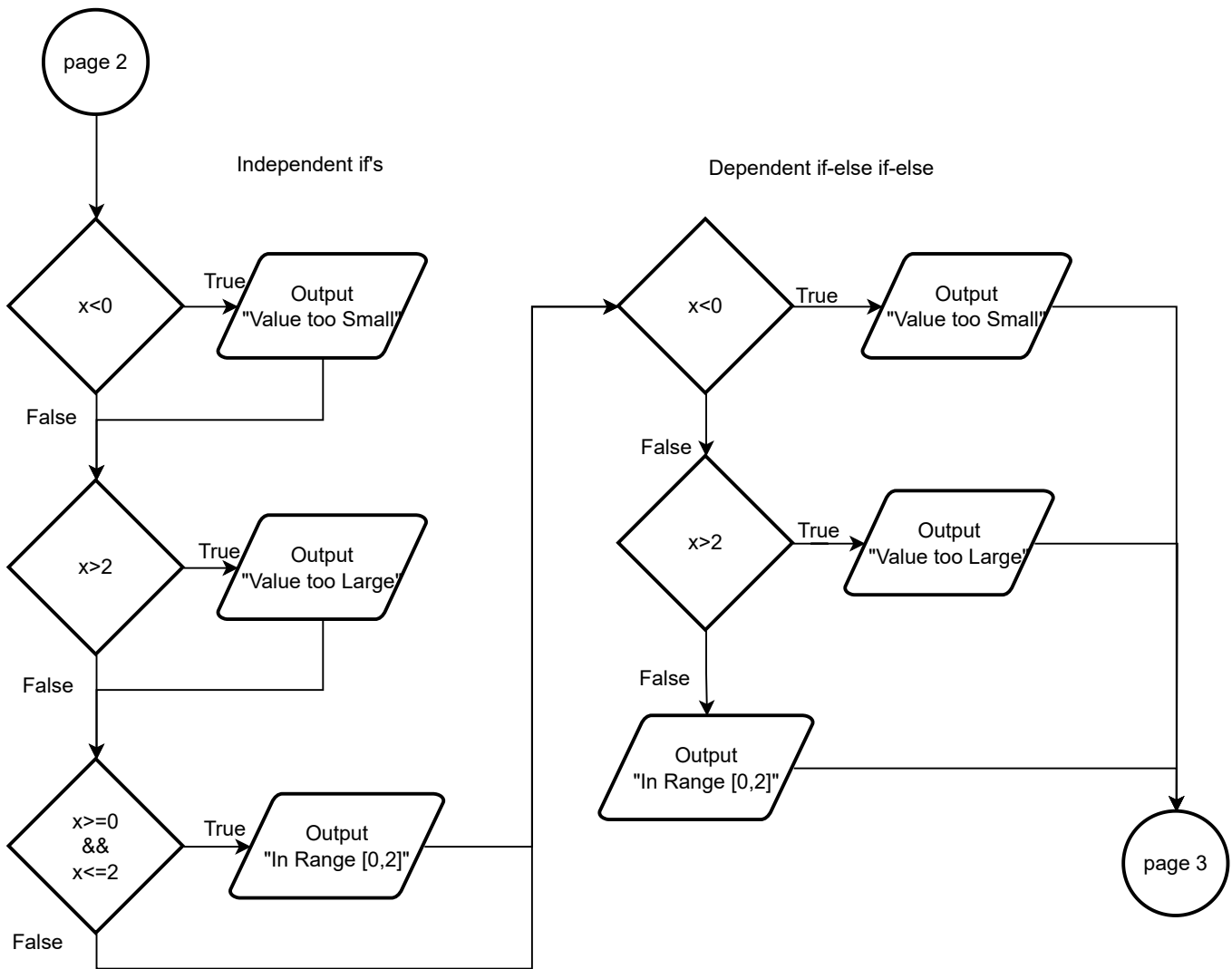
Nested Ternary Operator and a Switch Statement

(page 3)

Nested Ternary Operator

Switch Statement

x<0 — True → msg= "Value too Small"

False ↓

x>2 — True → msg= "Value too Large"

False ↓

msg= "In Range [0,2]"

Output msg

x==0 — True

x==1 — True

x==2 — True

Output "In Range [0,2]"

Output "Too small or too large"

(page 4)
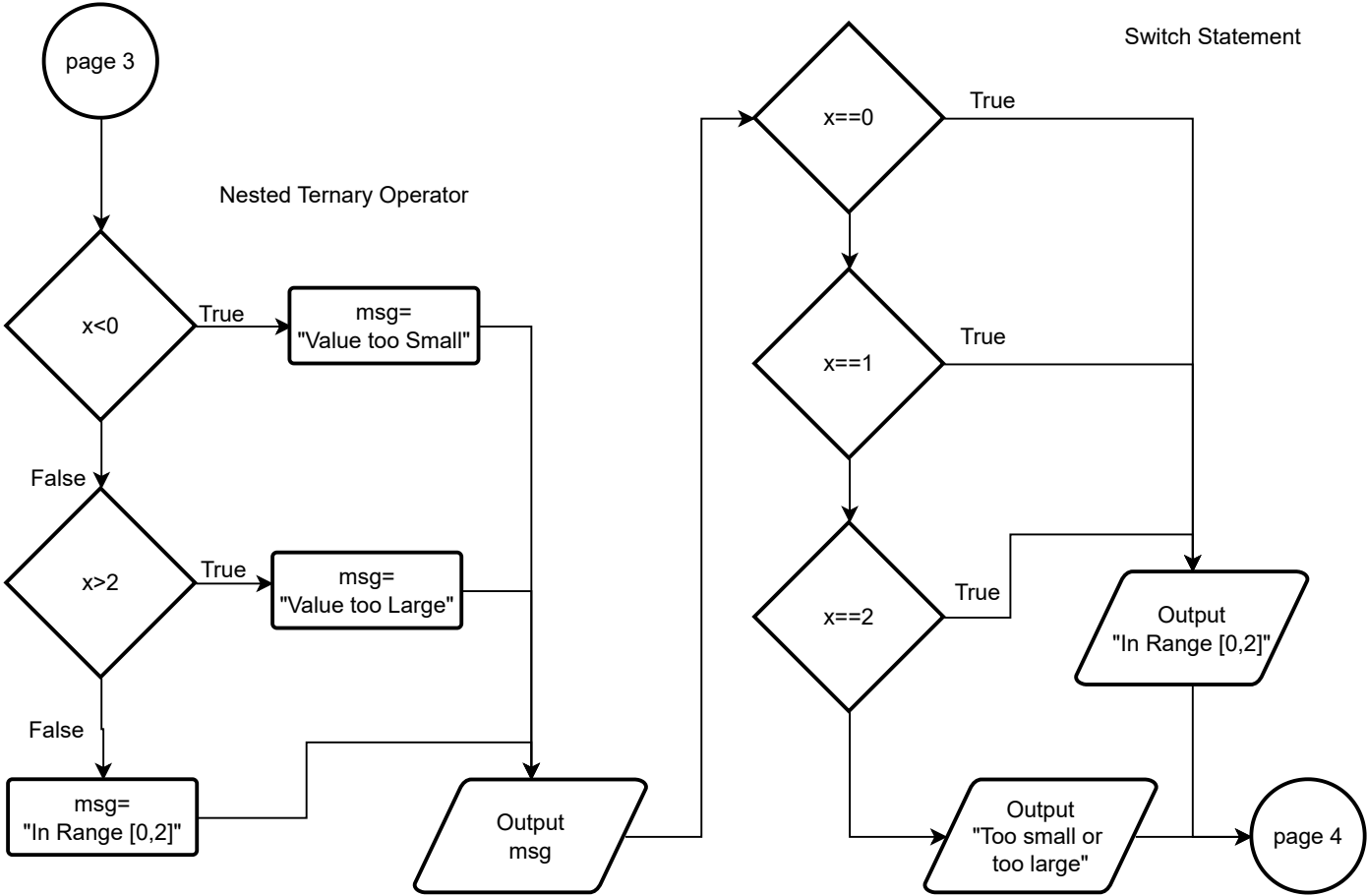
The 3 different Looping Constructs
For, Do-While and the While Loop

(page 4)

```
Declare and
Initialize
Out of Loop

floop=5
```

```
Declare and
Initialize
In For Loop

i=1
```

for-loop

i++

i<=floop → True → Output i

False

```
Declare and
Initialize
Out of Loop

dloop=5, i=1
```

do-loop

Output i

++i

i<=dloop → True

False

```
Declare and
Initialize
Out of Loop

wloop=5, j=1
```

while-loop

++j

j<=wloop → True → Output j

False

return 0
Exit from main