

structureintro [Protected View] - PowerPoint

File Home Insert Design Transitions Animations Slide Show Review View Storyboarding Tell me... Sign in Share

PROTECTED VIEW Be careful—files from the Internet can contain viruses. Unless you need to edit, it's safer to stay in Protected View. Enable Editing

1 Logic Structure - Focus on looping
Please use speaker notes for additional structure!

2

3

4

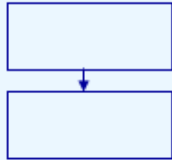
5

6

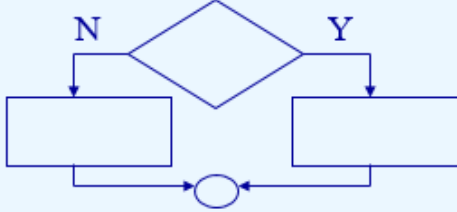
7

Programming logic involves three structures:

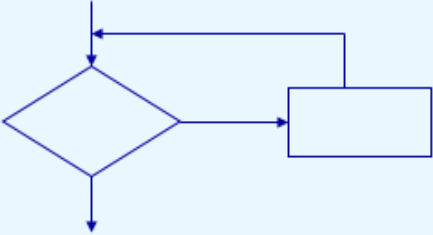
sequence structure



selection structure (conditions)



loop structure (iteration)



The three structures are all that is logically needed to write a procedural or structured program. All programming commands can be incorporated into these three basic structures.

Slide 2 of 7

Comments

11:00 AM 10/3/2017

structureintro [Protected View] - PowerPoint

File Home Insert Design Transitions Animations Slide Show Review View Storyboarding Tell me... Sign in Share

PROTECTED VIEW Be careful—files from the Internet can contain viruses. Unless you need to edit, it's safer to stay in Protected View. Enable Editing

DO WHILE LOOP

```

graph TD
    Start(( )) --> Condition{condition}
    Condition -- Y --> Processing[processing]
    Processing --> Condition
    Condition -- N --> Exit(( ))
  
```

Prior to processing

Processing in loop decision

DO WHILE LOOP:
 The while loop shown here tests a condition to see if the processing should be done.
 If the answer to the condition is YES, the processing box shown is executed.
 If the answer to the condition is NO, the processing box shown is not executed.

Might never process

This shows the DO while loop which is frequently used in programming to cause processing to be repeated until a specific condition is met. When the condition is met the processing will not be executed and control will drop out of the loop.

Slide 3 of 7

Comments

11:03 AM 10/3/2017

structureintro [Protected View] - PowerPoint

File Home Insert Design Transitions Animations Slide Show Review View Storyboarding

PROTECTED VIEW Be careful—files from the Internet can contain viruses. Unless you need to edit, it's safer to stay in Protected View. Enable Editing

DO UNTIL LOOP

```
graph TD; Start(( )) --> Processing[processing]; Processing --> Condition{condition}; Condition -- Y --> Processing; Condition -- N --> Exit(( ));
```

Always process at least once

DO UNTIL LOOP:
The while loop shown here executes the processing once and then tests a condition to see if the processing should be done. This means that the loop will always be executed once since it is executed before checking is done. Once the processing has been done once, further processing is determined by the answer to the condition. If the condition gets a YES, then the processing is executed again. If the condition gets a NO, then the processing is not executed again.

In this structure, the processing is done once and then the condition is checked to see if it should be done again. In this structure, the processing will always be done at least once since the check is after the completion of the processing.

Slide 4 of 7

Comments

77%

11:05 AM 10/3/2017

structureintro [Protected View] - PowerPoint

File Home Insert Design Transitions Animations Slide Show Review View Storyboarding

PROTECTED VIEW Be careful—files from the Internet can contain viruses. Unless you need to edit, it's safer to stay in Protected View. Enable Editing

DO WHILE LOOP

This example shows a do while loop where I am reading records from a file (getting input from a file). I want the processing to continue as long as there are records on the file. To do this, I am going to use an initializing read. I read the initial record and then I process a loop until the end of file (EOF) has been reached. To make this work, I always read or input another record at the end of the loop.

```

graph TD
    Start([Start]) --> Read[/Initializing Read/]
    Read --> NotEOF{Not EOF}
    NotEOF -- Y --> Process[Process]
    Process --> Read2[/Read/]
    Read2 --> NotEOF
    NotEOF -- N --> Stop([Stop])
  
```

Pseudocode:

```

start
input/read record
do while not EOF
  process
  input/read record
end while loop
stop
  
```

Note that I use the term initializing read, priming read is another term that can be used. When you are inputting a record you can use the word read, the word input, the word get or any word that implies retrieving a

Slide 5 of 7

11:07 AM 10/3/2017

structureintro [Protected View] - PowerPoint

File Home Insert Design Transitions Animations Slide Show Review View Storyboarding Tell me... Sign in Share

PROTECTED VIEW Be careful—files from the Internet can contain viruses. Unless you need to edit, it's safer to stay in Protected View. Enable Editing

DO WHILE LOOP

This example shows a do while loop controlled by a counter. I have determined that I want to continue the looping process while the counter is less than the stop point. When this condition is no longer true, I will exit the loop

Start point 1 Stop point 5

```

Pseudocode:
counter = startPoint
do while counter < stopPoint
  process
  increment counter
end while loop
  
```

```

graph TD
    Start([Start]) --> Init[counter = start point]
    Init --> Decision{counter < stop point}
    Decision -- Y --> Process[process]
    Process --> Increment[increment counter]
    Increment --> Decision
    Decision -- N --> Exit([Exit])
  
```

The looping structure where I want to use the **while loop** to do something a certain number of times, requires the following:

- **initialize the counter outside the loop to a specific start point**
- **test the counter to determine whether or not to enter the loop**
- **increment the counter inside the loop**

Counter
~~1~~
~~2~~
~~3~~
~~4~~
 5

In this slide, I am showing the specifics of the do while structure. I am using a counter to determine how many times I want to do a loop. I set the counter to the start point outside the loop. I then test the counter against the stop point to determine if I

Slide 6 of 7

Comments

11:10 AM 10/3/2017

- **initialize the counter outside the loop to a specific start point**
- **test the counter to determine whether or not to enter the loop**
- **increment the counter inside the loop**

When you write loops you need to initialize the control before the loop. So if counter is the control then I need to set up the memory variable counter and give it an initial value. Then the loop will contain a test of the control that will let the loop end. Finally I need to change the control within the loop, like adding 1 to counter.

structureintro [Protected View] - PowerPoint

File Home Insert Design Transitions Animations Slide Show Review View Storyboarding

PROTECTED VIEW Be careful—files from the Internet can contain viruses. Unless you need to edit, it's safer to stay in Protected View. Enable Editing

DO UNTIL LOOP

This example shows a do until loop controlled by a counter. I have determined that I always want to process once and that I want to continue the looping process until the counter equals the stop point. When this condition is true, I will exit the loop

Pseudocode:
Startpoint stoppoint
 counter = startPoint
 do
 process
 increment counter
 until counter = stopPoint

Counter
 1
 2
 3
 4
 5

The looping structure where I want to use the **until loop** to do something a certain number of times, requires the following:

- **initialize the counter outside the loop to a specific start point**
- **process**
- **increment the counter inside the loop**
- **test the counter to determine whether or not to loop again**

```

graph TD
    A[counter = start point] --> B[process]
    B --> C[increment counter]
    C --> D{counter not = stop point}
    D -- Y --> B
    D -- N --> E[ ]
  
```

In the do until loop, I will always process at least once. I think it is clearer when writing the pseudocode, to show the until condition at the bottom of the loop rather than at the top.

Title: Oct 3-11:13 AM (7 of 11)

assignplay1 [Protected View] - PowerPoint

File Home Insert Design Transitions Animations Slide Show Review View Storyboarding

PROTECTED VIEW Be careful—files from the Internet can contain viruses. Unless you need to edit, it's safer to stay in Protected View. Enable Editing

1 Loop Assignment
2
3
4
5
6
7

Start

Initializing
Read a record

Not EOF

Y

Calculate profit

Set up line for report

Write line on report

Read a record

N

Stop

Problem #1: I want to produce a report using the logic shown. The report should contain ItemNo, ItemName, Price, Cost and Profit. Profit is the difference between Price and Cost.

95.89
75
20.89 17.99

ItemNo	ItemName	OnHand	OnOrder	ReorderPt	Cost	Price	VendorNo
1111	Oak	5	10	50	75	95.89	123
1212	Red Maple	26	22	25	100	117.99	124
1234	Sugar Maple	45	15	50	45	65.89	123
1245	Hemlock	10	17	50	35	49.99	125
1256	Blue Spruce	10	29	25	75	110.99	127
1267	White Pine	25	50	50	35	45.75	125
1275	Black Pine	7	12	25	70	105.98	127
1290	Birch	11	0	25	80	110.75	124
2000	Dogwood	12	20	45	75	95.99	124
2012	Cherry	35	10	25	60	75.75	124
2036	Elm	50	25	25	75	90	123

1111 Oak 95.89 70 20.89
1212 RM 117.99 100 17.99

Slide 3 of 10

Comments

77%

assignplay1 [Protected View] - PowerPoint

File Home Insert Design Transitions Animations Slide Show Review View Storyboarding

PROTECTED VIEW Be careful—files from the Internet can contain viruses. Unless you need to edit, it's safer to stay in Protected View. [Enable Editing](#)

Program:

```

start
input/read record
do while not EOF
    process ()
end while loop
stop

```

```

process ()
    if VendorNo = "124"
        if OnOrder > 10
            setup line for report
            write line for report
        else
            if Price > 100.00
                setup line for report
                write line for report
            end if
        end if
    end if
    input/read a record
end process

```

ItemNo	ItemName	OnHand	OnOrder	ReorderPt	Cost	Price	VendorNo
1111	Oak	5	10	50	75	95.89	123
1212	Red Maple	26	22	25	100	117.99	124
1234	Sugar Maple	45	15	50	45	65.89	123
1245	Hemlock	10	17	50	35	49.99	125
1256	Blue Spruce	10	29	25	75	110.99	127
1267	White Pine	25	50	50	35	45.75	125
1275	Black Pine	7	12	25	70	105.98	127
1290	Birch	11	0	25	80	110.75	124
2000	Dogwood	12	20	45	75	95.99	124
2012	Cherry	35	10	25	60	75.75	124
2036	Elm	50	25	25	75	90	123

Problem #6: Produce this report. The output should show ItemNo, ItemName, OnHand, OnOrder and Price.

Slide 8 of 10

Comments

77%

Programming: Logic, Design and Implementation CIS120/17

About This Course:

Read Me First
Introduction to CIS120/17
Syllabus
Withdrawal Policy as of Fall
2017

Site Resources:

Notes and Handouts
Examples
Programs
Presentations
SmartBoard Presentations
Audio/lectures

Weekly Schedule:

Weekly schedule
Assignment summary

Links:

Links & Tutorials

The information at this site is in draft form and is the property of Priscilla Grocer. The information may not be duplicated or distributed without her permission.
©Copyright 2017

Week #5
Week of
October 2nd

e are trying different times, Erik will be available Monday from 2-3, Tuesday from 2-4 and Thursday from 4:30-6:30. I will come and go. I need to cancel my office hour on Tues at 2, I have a meeting. I will have an office hour Monday at 10. I will also be around later on Tues and Thurs afternoon. Check K101 or my office. I strongly suggest you set up some study groups and meet when Erik is available to answer questions.

We will look at logic some more this week.

Assignments:

We worked on this in class.

Logic PowerPoint assignment

Flowchart and Pseudocode assignment This is a quiz - however it is not due until next Thursday (October 12th) when homework is due - you need to work on it individually because it is a quiz (that also means no resubmit). This quiz also counts more than the quick quizzes - guess I should have called it a test. It counts like a high-end homework. Check back!

We are trying different times, Erik will be available Monday from 2-3, Tuesday from 2-4 and Thursday from 4:30-6:30. I will come and go. I have an office hours Mon at 10 in my office. I strongly suggest you set up some study groups and meet when Erik is available to answer questions.

Stem Activity

This is Enrollment Verification week and I have to verify you are in the class. If you have not passed in work or the email a week verifying you are in the class, then you need to contact me now to make sure you are not withdrawn from the class.

Reading - there is not a required text but there is reading that will add to your knowledge base and that is required:

If you bought the optional text, read chapter 1 (you might note that at Amazon much of the first chapter is available for [Just Enough Programming Logic and Design](#) in the Look inside feature). Might help you decide about the book.

Compiler definition

Compiler vs interpreter

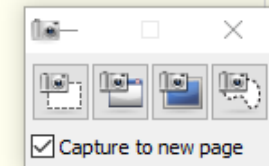
Types of software

Memory and RAM

Programming Process

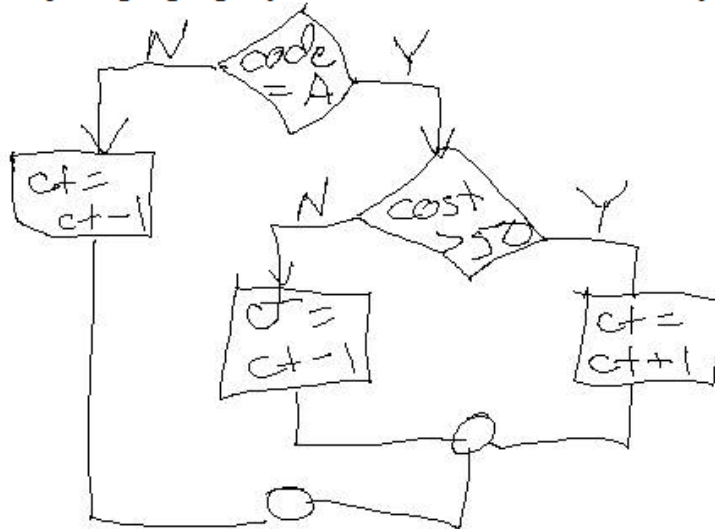
Examples: Pseudocode and Flowcharts

Pseudocode and Flowcharts



Logic Flowchart and Pseudocode Quiz

This quiz is going to give you a flowchart for a condition and ask you to answer questions about it.



~~10~~
ct

A
code

25
cost

[Comment on the test](#)

Problem #1: Assume the following:

- 1 will be put into the memory variable ct
- A will be put into the memory variable code
- 25 will be put into the memory variable cost

After following the logic in the flowchart, what will be in the memory variable ct?

Problem #2: Assume the following:

- the memory variable ct will contain your answer from problem #1
- A will be put into the memory variable code
- 100 will be put into the memory variable cost

After following the logic in the flowchart, what will be in the memory variable ct?

Problem #3: Assume the following:

