

CIS120/17 Course Page

www.pgrocer.net/PFGCIS120.html

**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 #4  
 Week of  
 September 25th

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** Please read...  
**Memory and RAM**  
**Programming Process**  
**Examples: Pseudocode and Flowcharts**  
**Pseudocode and Flowcharts**  
**Data types**  
 I also want to include some information on skills, so please listen to this video and read this handout about note taking.  
**Note taking video**  
**Cornell method of note taking** is the specific one to read. I also suggest checking out the other links on this page.  
 We went over SQL last week. Here are some examples you may find helpful:  
**Some SQL examples** Let me know if you find a typo. [Examples of SQL to look at](#)  
 This week, we will go over some of the principles from the readings above and start looking at logic. We will work on the practice exercise below.  
***Practice exercise:***  
**Practice exercise**  
**Practice continued**  
**Presentation on using pseudocode to play computer**  
**Separate speaker notes to accompany presentation on using pseudocode to play computer**  
 Check back!

I will be available Monday at 10 in my office K112 and Tuesday and Thursday at 2 in K101 (hopefully). Erik will be available at 2 on Mon, Tues and Thurs and he will also be available one night at 6 - need to check with him. If you need help with something, now would be the time to ask your questions!  
**Directions for downloading Access are located under Notes and Handouts. Using Office**

start

www.pgrocer.net/Cis17/inclass/loopifplay.html

SMART Ink

In Class Exercise on loops and if statements:  
 These are unrealistic problems, but they help to make sure that you understand the flow of logic and can follow it through to completion.

Problems #1:

```

start
  firstNum = 100
  secondNum = 5
  thirdNum = 12
  the Result = 0
  do while secondNum <= thirdNum
    if firstNum > 500
      firstNum = firstNum * 2
      secondNum = secondNum + 1
    else
      firstNum = firstNum + 100
      thirdNum = thirdNum - 1
    end if
  end while loop
  theResult = firstNum + secondNum + thirdNum
  display theResult
end
  
```

set memory variables  
 Initializing house keeping  
 processing in loop here  
 wrap up ending

Problem #2:

```

start
  firstNum = 25
  secondNum = 50
  thirdNum = 75
  fourthNum = 100
  ct = 0
  workAns = 0
  theAns = 0
  do while ct < 6
    workAns = firstNum + secondNum
    if workAns > thirdNum
      fourthNum = fourthNum / 2
    end if
    ct = ct + 1
  end while
end
  
```

loop  
 set up control  
 check  
 change

Program

- ① Sequence
- ② Selection decisions  
if
- ③ repetition loop

Windows taskbar: 12:43 PM 9/26/2017

start

www.pgrocer.net/Cis17/inclass/loopifplay.html

**In Class Exercise on loops and if statements:**  
 These are unrealistic problems, but they help to make sure that you understand the flow of logic and can follow it through to completion.

**Problems #1:**

```

start
  firstNum = 100
  secondNum = 5
  thirdNum = 12
  the Result = 0
  do while secondNum <= thirdNum
    if firstNum > 500
      firstNum = firstNum * 2
      secondNum = secondNum + 1
    else
      firstNum = firstNum + 100
      thirdNum = thirdNum - 1
    end if
  end while loop
  theResult = firstNum + secondNum + thirdNum
  display theResult
end
  
```

*Handwritten notes for Problem #1:*  
 - set memory variables (circled around initialization)  
 - Initializing house keeping (circled around initialization)  
 - processing in loop here (circled around the while loop body)  
 - wrap up ending (circled around the end of the loop and result display)

**Problem #2:**

```

start
  firstNum = 25
  secondNum = 50
  thirdNum = 75
  fourthNum = 100
  ct = 0
  workAns = 0
  theAns = 0
  do while ct < 6
    workAns = firstNum + secondNum
    if workAns > thirdNum
      fourthNum = fourthNum / 2
    end if
  end while
end
  
```

*Handwritten notes for Problem #2:*  
 - loop (underlined)  
 - set up control  
 - check  
 - change

firstNum = firstNum \* 2 means take the contents of firstNum and multiply by 2 and ASSIGN it back to firstNum.  
 The = is an assignment sign.

Program

- ① Sequence
- ② Selection decisions  
if
- ③ repetition loop

Windows taskbar: 12:43 PM 9/26/2017

**In Class Exercise on loops and if statements:**

These are unrealistic problems, but they help to make sure that you understand the flow of logic and can follow it through to completion.

**Problems #1:**

```

start
  firstNum = 100
  secondNum = 5
  thirdNum = 12
  theResult = 0
  do while secondNum <= thirdNum
    if firstNum > 500
      firstNum = firstNum * 2
      secondNum = secondNum + 1
    else
      firstNum = firstNum + 100
      thirdNum = thirdNum - 1
    end if
  end while loop
  theResult = firstNum + secondNum + thirdNum
  display theResult
end
  
```

<u>First Num</u>	<u>second Num</u>	<u>third Num</u>
100	5	12
200	6	11
300	7	10
400	8	9
500		8
600		7

the Result  
4815

4815

**Problem #2:**

```

start
  firstNum = 25
  secondNum = 50
  thirdNum = 75
  fourthNum = 100
  ct = 0
  workAns = 0
  theAns = 0
  do while ct < 6
    workAns = firstNum + secondNum
    if workAns > thirdNum
      fourthNum = fourthNum / 2
    end if
  end while
end
  
```

= assignment sign

100
2400
4800

start

```

end if
end while loop
theResult = firstNum + secondNum + thirdNum
display theResult
end

```

Problem #2:

```

start
firstNum = 25
secondNum = 50
thirdNum = 75
fourthNum = 100
ct = 0
workAns = 0
theAns = 0
do while ct < 6
  workAns = firstNum + secondNum
  if workAns > thirdNum
    fourthNum = fourthNum / 2
    firstNum = firstNum * 3
    secondNum = firstNum - secondNum
    thirdNum = thirdNum * 2
  else
    firstNum = firstNum + 10
    secondNum = secondNum + 10
    thirdNum = thirdNum + 10
  end if
  ct = ct + 1
end do
theAns = firstNum + secondNum * thirdNum + fourthNum
display theAns
end

```

Handwritten notes and calculations:

ct  
0  
+ 1  
~~2~~  
3  
~~4~~  
5  
6

First Num  
25  
35  
~~105~~  
115  
~~125~~  
135  
~~1105~~

Second Num  
50  
~~60~~  
45  
55  
65  
75  
330

Third Num  
75  
~~85~~  
170  
180  
190  
200  
400

Fourth Num  
100  
50  
25  
150  
170  
190  
210

workAns  
0  
75  
95  
150  
170  
190  
210

the Ans  
0  
132430

Handwritten calculations:

$$\begin{array}{r}
 330 \\
 + 400 \\
 \hline
 132430
 \end{array}$$

$$\begin{array}{r}
 400 \\
 + 25 \\
 \hline
 425
 \end{array}$$

$$\begin{array}{r}
 132430 \\
 + 25 \\
 \hline
 132455
 \end{array}$$

File Tools View ifexampleS10-1 (Protected View) - Word SMART Ink

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

Practice exercise:

Payroll file: (each record is payRecord)

idno	name	payHr	regHrs	ovtHrs
1111	John Doe	20	40	10
1234	Ann Costa	25	40	0
2222	Linda Ryan	30	40	20
2345	Bob Smith	30	35	10
EOF				

```

start
read payRecord
do while not endOfFile
  if ovtHrs > 0
    regPay = payHr * regHrs
    ovtPay = payHr * ovtHrs * 1.5
    totPay = regPay + ovtPay
  else
    totPay = payHr * regHrs
  end if
  display totPay
  read payRecord
end do while loop
stop program

```

Handwritten calculations:

- Left side (circled):** 1100 (blue), 1000 (red), 2100 (green), 1500 (black)
- Middle:**
  - reg Pay: 800 (blue), 1200 (green), 1050 (black)
  - ovt Pay: 300 (blue), 900 (green), 450 (black)
- Right side:**
  - tot Pay: 1100 (blue), 1000 (red), 2100 (green), 1500 (black)

Arrows indicate data flow from the table to the code and then to the calculations.

File Tools View ifexampleS10-1 (Protected View) - Word SMART Ink

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

Practice exercise:

Payroll file: (each record is payRecord)

idno	name	payHr	regHrs	ovtHrs
1111	John Doe	20	40	10
1234	Ann Costa	25	40	0
2222	Linda Ryan	30	40	20
2345	Bob Smith	30	35	10
EOF				

```

start
read payRecord
do while not endOfFile
  if ovtHrs > 0
    regPay = payHr * regHrs
    ovtPay = payHr * ovtHrs * 1.5
    totPay = regPay + ovtPay
  else
    totPay = payHr * regHrs
  end if
  display totPay
  read payRecord
end do while loop
stop program

```

*Handwritten notes:*

- Initializing read* (red text, pointing to the first 'read payRecord' line)
- all other records read here* (red text, pointing to the 'do while' loop)
- reg Pay* calculations:
  - 800
  - 1200
  - 1050
- ovt Pay* calculations:
  - 300
  - 900
  - 450
- tot Pay* calculations:
  - 1100
  - 1000
  - 2100
  - 1500
- A blue circle on the left contains the values: 1100, 1000, 2100, 1500.
- Arrows from the table rows point to the corresponding calculations.

Windows taskbar: 1:45 PM 9/26/2017

File Tools View ifexampleS10-1 (Protected View) - Word

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

Practice exercise:

Payroll file: (each record is payRecord)

idno	name	payHr	regHrs	ovtHrs
1111	John Doe	20	40	10
1234	Ann Costa	25	40	0
2222	Linda Ryan	30	40	20
2345	Bob Smith	30	35	10
EOF				

```

start
read payRecord
do while not endOfFile
  if ovtHrs > 0
    regPay = payHr * regHrs
    ovtPay = payHr * ovtHrs * 1.5
    totPay = regPay + ovtPay
  else
    totPay = payHr * regHrs
  end if
  display totPay
  read payRecord
end do while loop
stop program

```

*Handwritten notes:*

- Initializing read* (red text, pointing to the first 'read payRecord' line)
- all other records read here* (red text, pointing to the 'do while' loop)
- reg Pay* calculations: 800 (blue), 1200 (green), 1050 (black)
- out Pay* calculations: 306 (blue), 900 (green), 450 (black)
- tot Pay* calculations: 1100 (blue), 1000 (red), 2100 (green), 1500 (black)
- A blue circle highlights the 'tot Pay' calculations on the left.
- A blue arrow points from the 'EOF' row of the table to the 'read payRecord' line.
- A red arrow points from the 'read payRecord' line to the 'do while' loop.
- A green arrow points from the 'do while' loop to the 'display totPay' line.
- A black arrow points from the 'display totPay' line to the 'read payRecord' line.



File Tools View ifexampleS10-1 (Protected View) - Word

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

Practice exercise:

Payroll file: (each record is payRecord)

idno	name	payHr	regHrs	ovtHrs
1111	John Doe	20	40	10
1234	Ann Costa	25	40	0
2222	Linda Ryan	30	40	20
2345	Bob Smith	30	35	10
EOF				

```

start
read payRecord
do while not endOfFile
  if ovtHrs > 0
    regPay = payHr * regHrs
    ovtPay = payHr * ovtHrs * 1.5
    totPay = regPay + ovtPay
  else
    totPay = payHr * regHrs
  end if
  display totPay
  read payRecord
end do while loop
stop program

```

*Handwritten notes:*

- Initializing read (red arrow pointing to the first read payRecord line)
- 1100 (blue), 1000 (red), 2100 (green), 1500 (black) (circled in blue)
- reg Pay 800 (blue)
- out Pay 306 (blue)
- tot Pay 1100 (blue), 1000 (red), 2100 (green), 1500 (black) (vertical list)
- 900 (green), 450 (black) (vertical list)
- 1200 (green), 1050 (black) (vertical list)
- all other records read here (red arrow pointing to the loop body)

100%