**Chapter 1 – Week 3 – Exercises**

Exercises #1 – page 5

**1. List three common types of computing agents.**

1-Humans

2-Cash registers

3-Laptop computers

4 (bonus)-Echo dot

**2. Write an algorithm that describes the second part of the process of making change (counting out the coins and bills)**

Assumptions:

1. We know how to add, divide and compare monetary values.
2. We are using the United States currency system and have all common coins (pennies, nickles, dimes, and quarters) and bills ($1, $5, $10, and $20) available .
3. The purchase is less that $100 and only a single bill is received.

Steps:

1. Receive the payment from the user and place it on the bill storage face up.
2. Start with the amount of the purchase and keep a running tally of the purchase price plus the change you holding to give to the customer.
3. Determine if the cents portion of the current tally (from now on simply referred to as the tally) is divisible by 5. If not, add penny(ies) to the change and tally until it is.
4. Now determine if your tally is divisible by 25. If not, add dimes to the change and tally until adding more would go over the next value that is divisible by 25 (quarter value). Then, add a nickel if needed to reach the nearest quarter value.
5. Determine if you tally is divisible by a dollar. If not, add quarters until it is.
6. Stop and congratulate yourself on a job well done. This separates the coin portion of our change calculation. We will now move on to dollars.
7. Determine if your tally is divisible by $5. If not, add $1 bills until it is.
8. Determine if your tally equals the bill received from the customer. If not, add a $5 bill if necessary to make your tally divisible by $10.
9. Again, check if your tally equals the amount received. If not, add $20 bills as long as it would not cause the tally to exceed the amount received.
10. If the tally is still not correct, you will need to add a $10 bill.
11. Give the change to the customer making sure to count it back and receive agreement that it is the proper amount.
12. Place your bill into the correct location in your drawer, close it. It is time to start over with the next customer.

**3. Write an algorithm that describes a common task, such as baking a cake or operating a DVD player.**

Common Task: How to read a book.

Assumptions:

1. I own the book I want to read.
2. The book is a hard copy and not digital.
3. I am the one reading the book and thus know how the genre of the book was classified.
4. I know how to manipulate a book for reading.

Steps:

1. Decide which book I want to read.
2. Determine the genre of this book.
3. Go to the appropriate section of my bookshelves that contains the books in this genre.
4. Search through the genre’s books one by one until the correct book is found.
5. Take book off of the shelf and place it into hands.
6. Carry book to the nearest chair and sit down.
7. Ensure the text on the cover is in the desired direction. If not, rotate the book until it is.
8. Look for a bookmark indicating where I previously stopped reading. If one is found, open the book to the bookmark, remove it, begin reading at the location in the book, and skip ahead to step 11.
9. Open the book by turning the cover to the left.
10. Investigate each page and turn to the next until the beginning of the book is reached. (Use your judgment as to where that is.)
11. Read each page and turn to the next page until the end of the book is reached.
12. If you must stop due to a nagging wife (or any other reason), place a book mark in the book and return it to the shelf.

**4. Describe an instruction that is not well defined and thus could not be included as a step in an algorithm. Give an example of such an instruction.**

A poorly described instruction is one that non-specific. This can be because it is too general, the agent does not know enough about the topic to understand the instruction, or perhaps the instruction is impossible to accomplish.

A too general statement could be something like “prevent a war”. How in the world is someone supposed to do that. However, prevent a war could be broken into pieces such as negotiation, building a strong military force, and/or praying.

A statement such as “balance your checkbook” is easy enough for me. However, my kids promise me they will never write a check, look puzzled at the idea of a checkbook, and have never had need to do this. Therefore, they lack the knowledge necessary to accomplish the task without research. And once they have the knowledge, they will need to create an algorithm to do it at that time.

The example of determining the result of dividing 8 apples among 0 people is an example of an impossible task. Mathematics has defined the result of division by 0 as meaningless. Therefor, asking someone to do this will not result in a fruitful answer.

**5. In what sense is a laptop computer a general-purpose problem-solving machine?**

A laptop is a general-purpose problem solver in the sense that it is capable of solving any problem that is solvable with a keyboard, pointing device, monitor, and perhaps printer that you can think of. Doing so requires having a knowledgeable person to program it, the understanding of the problem to solve the problem, and a problem that is solvable in a reasonable amount of time.

**6. List four devices that use computers and describe the information that they process (Hint: Think of the inputs and outputs of the devices)**

4 devices that use computers and the information they process are:

Sirrius XM radio receiver in my truck:

The radio receiver receives user input in the form of channel selection as well as time, date, and other settings. It also receives signal input from the satellite or other radio source.

The radio relays information to the user via the speakers playing music or broadcasting voices according the information on the radio signals.

Kindle e-reader:

The Kindle receives user input in the form of book selection, page turning, book purchasing, etc. It receives it via its touch screen interface. Additional input comes via the Internet or file transfer. This is how the book contents are passed to the user.

The Kindle’s output is obviously the contents of the book the user reads.

Calculator:

A simple device receiving input via the keypad through button presses. The user enters numeric and mathematical expressions such as addition (+) and multiplication (\* or x).

It delivers output via a small 8 to 12ish digit screen that displays numerical data.

Router:

The router at my house accepts user input via an app on my phone. This app allows one to set the security password for the wifi it provides, prevent certain devices from accessing the network, etc. via its mobile interface. Many routers offer a simple web interface that can be accessed from web browsers withing the network for similar inputs. It also receives data packets from inside and outside the network.

As far as output, the router allows (or denies) access for a device to communicate with other devices reachable from the router. It receives packets from users on the network as input and passes them out to the network (output). The process also works in reverse.

Exercises #2 – page 9

1. **List two examples of input devices and two examples of output devices**

Two example of input devices are the keyboard and mouse. As I bonus I offer the microphone.

1. **What does the central processing unit (CPU) do?**

The CPU accepts electronic signals as inputs. These inputs describe an action to perform such as add two numbers together or place this value into register 2. The signals travel through the unit being manipulated into the correct result (almost all of the time). The result is, of course, also in electronic signals. These input and output signals can be transferred into memory (RAM) or to disk. These signals represent on/off states or the 1s and 0s of binary code.

1. **How is information represented in hardware memory?**

The information is stored in memory ultimately as binary values. These values are formed into groups of 8 called bytes. On today’s computers these bytes are grouped together into words. A 32-bit machine is also a 4 byte word machine.

1. **What is the difference between a terminal-based interface and a graphical user interface?**

Terminal based interfaces are in text, easy (ish) to code for, but not too pretty. GUIs allow the user to interact with a vibrant, colorful, image oriented interface that easier and often more fun for the user. This comes at the expense of being more difficult to program.

1. **What role do translators play in the programming process?**

Translators (or compilers) convert a programmers code from the “high” level programming languages of today that offer phenomenal features that make writing programs easier and easier. (Alas, not easy just easier). These commands such as print “Hello World” or x = 3 + z \* 4 are not readily understood by our CPU. So, someone has to convert the human programmer speak (aka: code) into computer speak (aka: binary machine code). Compilers are what we call programs that do this for us. Sometimes the compiler is responsible to convert our code into actual machine code for a given processor, however, more and more compilers are translating into code for a virtual machine. This way the compiled program is the same regardless of the machine it runs on. On the other hand, another program (the virtual machine for the language) must be written for each type of processor in the world.

Exercises #3 – page 29

1. **Describe what happens when the programmer enters the string “Greetings!” in the Python shell.**

When the programmer enters a string in quotes such as “Greetings!” the Python shell simply displays in back to the user in single quotes. This is because the Python shell is an example of a REPL environment. REPL = Read – Execute – Print Loop. The shell reads the command, then it executes. The results of the command are the string that was typed. It then prints the result to the screen. Since it is a string, Python “wraps” the string in quotations marks.

1. **Write a line of code that prompts the user for his or her name and save the user’s input in a variable called name.**

name = input( "Como te llamas? " )

1. **Answer the question: What is a Python script?**

A Python script is a text file with a collection of Python commands to be run in sequence. They can be executed from the operating system shell via its command line.

1. **Explain what goes on behind the scenes when your computer runs a Python program.**

When a Python program/script is run:

* 1. The script is loaded and read line by line
  2. The syntax checker confirms the Python statement is legal and follows the correct syntax. Syntax errors are reported at this time.
  3. The code is then translated into byte code and passed to the Python Virtual Machine.
  4. The Python VM executes the supplied code.

Exercises #4 – page 30

1. **Suppose your script attempts to print the value of a variable that has not yet been assigned a value. How does the Python interpreter react?**

When Python encounters a variables that has not yet been assigned a value, it will result in an error. The message will be similar to this one:

>>> x = y \*\* 3

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

NameError: name 'y' is not defined

1. **Miranda has forgotten to complete an arithmetic expression before the end of a line of code. How will the Python interpreter react?**

When an arithmetic expression is not terminated before the end of a line, Python will, depending on the circumstances, create a syntax error. If Miranda has “3 +” on a line and pressed enter, the syntax error is seen. However, if she types “6 + ( 7 \*” and presses enter, Python will dutifully read the next line expecting a “)”. If she then enters “3 )” on the next line she will see the result 27.

1. **Why does Python code generate fewer types of syntax errors than code in other programming languages?**

Fewer syntax error are encountered in Python because Python has intentionally tried to create as simple and straight forward a syntax as possible. It also uses a more English friendly syntax than earlier languages did.