Term 1, Assignment 1 - Workbook

Jye Calder-Mason

Q1. Identify and explain common and important components and concepts of web development markup languages

Markup is the building blocks to the web, its a form of language code that allows the web to be communicated by code via different types of instructions. Markup is generally used to edit how a web page is appeared via a programing language.

Each part of the code modifies the end product of the code, so changing something small can change a lot in the long run. In web-development the main three 'Markup Languages' are HTML, XML and XHTML. When developing on the web, you would generally use Hypertext markup language 'HTML'. You can customize anything via this language, from visual design, web friendlyness, make interactions simple, or to the more simplicity of how big or small you wanted your text or photos. Whichever way you desire your website to look, HTML is the code for you.<

Q2. Define the features of the following technologies that are essential in terms of the development of the internet:

- packets
- IP addresses (IPv4 and IPv6)
- routers and routing
- domains and DNS

Explain how each technology has contributed to the development of the internet.

Packets:

Packets are small little unit of data, A packets is made up of controlled information and data, they're normally 1500 bytes of data each. Inside that data is the sender's IP address and the recievers IP address. These packets are thrown through the internet to communicate with each other to produce information for e.g emails, websites and general internet purposes.

Packets have changes the game for internet developement, because the use of splitting up the controllable data and information into packets has made it incredibly easy to communicate which route it will take and making it way easier to read.

Packets are like the Auspost of the internet. They have an address were they are going, a message or gift, they have a sender and a reciever, if something goes wrong it comes up with an error. Without packets the internet would have no hope.

IP addresses (IPv4 and IPv6):

An IP address is a identifier for devices or objects connected to the network. Each device has its own unique address consisting of a 32bit binary number, or in IPv4 it would be four eight binary numbers displaying the IP.

Internet Protocol version 4 (IPv4)is a 32bit address scheme that swaps packets between the link layer network via ethernet. IPv4 is the core protocol for internet traffic. When IPv4 came out, so little people needed it, so it worked smoothly, but as the world has grown the limited 32bit binary code (cap at 4.3 billion) for the IP has run out, thus needing a newer version IPv6.

So saying that IPv6 is here to cover for the limited space IPv4 had. All of the previous hurdles version 4 had has been updated in 6, including an upgrade from 32bit to 128bit address which now makes the cap at (340quatrillion). IPv4 is a numeric based address system whereas IPv6 is a alphabetic system.

Routers and routing:

A router is a brilliant piece of technology that sends and receives packets. Routing is method of moving traffic in a network.

Routing is the method of transport for your router to transfer and recieve packets. The routing will always take the most efficient route, if the traffic is busy on the way to your local internet hub, its generally because the routing has taken longer than expected, therefore making it slow.

A router is constantly running the link state algorithim to figure out which efficient route it has to path to the destination. A link state is home base for the router, and the algorithm is constantly solving the more efficent way through all the other link layers. This algorithm has contributed to the developement to the internet by allowing information and data to be transfered the fasted and most efficient way.

Domains and DNS

A domain name is one of two things, the name of said url 'coderacademy.com.au' and the IP version that the DNS (Domain Name Service) communicates with '255.255.255.0' etc. From the outside in, when viewing a website all you will see is the domain name 'coderacademy.com.au' but in reality its actually being called '255.255.255.0' via the DNS.

If all domain names were a bunch of numbers, nobody would remember the address to other domains. The domain name allows you to type in 'coderacademy.com.au' and skip the struggle of typing in a bunch of numbers that you'll most likely forget. Everytime you search for a website the DNS keeps track of the IP addresses assosiated with the name, and translates that to the website. This has helped with the developement of the internet by making it easier to search the web with basic website domains.

Q3. Define the features of the following technologies that are essential in terms of the development of the internet:

- TCP
- HTTP and HTTPS
- web browsers (requests, rendering and developer tools)

Explain how each technology has contributed to the development of client and server communication over the internet (50 - 150 words for each technology)

TCP

Transmission Control Protocol (TCP) is a model designed to standardise computer networking. TCP defines how applications can create communication channels and also manages how a message is broken down to be transmitted. TCP is made up of five different layers, application, transport, network, data link and physical.

Everytime your computer or device transmits data throughout ethernet it will go through these five stages, either side of the transmission. It works as a pyramid adding data from each layer. It starts of at the application (HTTP/FTP/SMTP), then transports through TCP or UDP to then checking the connection through your IP via the router, then moves making your way into the ethernet to be sent off and done in reverse to the reciever, while taking off layers to make it back to the application. This method has contributed to the developement of communication over the internet.

HTPP and HTTPS

Hyptertext Transfer Protocol (HTTP) and Hypertext Transfer Protocol Secure (HTTPS) is a network protocol that enables connection to a host, webpage etc. Hypertext being the the content you're viewing on a web browser, this is the method of transfering between a server and a web server on the internet. When you click on a webpage it sends HTTP to the web server which then runs a application to process and send the HTTP back to the webpage.

When sending a request through HTTP to the server or webserver its unsecure, making it very available for anybody to view the data, but when connecting via HTTPS the connection is encrypted for nobody to view, making it safer to connect. This has contributed to the development of client and server communication by allowing the people to be able to request data from there computer to be displayed onto there home device.

web browsers (requests, rendering and developer tools)

A web browser is an application on the internet that gathers information to display or function abilities on the internet. There are multiple web browsers all with there own set of engines, each engine not working on other browsers, so the HTML and CSS code may be different if you looked between them. Internet explorer uses an engine called Trident, Google Chrome uses Blink, and Mozilla Firefox uses Gecko. How it all works is when you type in a HTTP or HTTPS url the browser requests the data to display onto the webpage, after its recieved the data is then rendered into a visual display on the browser. The developer tools allows the user inspect code via the browser, the code being HTML/CSS/JAVA/, this allows the developer to create, test and debug software before releasing there product. Web browsers have made a huge contribution to client and server communication by allowing users to connect to web in a smart and easy way. Also allowing developers to create and produce helpful products to make the whole experience better.

**** Q4. Identify THREE data structures used in the Python programming language and explain the reasons for using each.****

Data structures are a method of designating data to one object or another. Structured in; adding, deleting, listing the data in a manner. Data structures The three main structures, Ill be using today are; Lists, Dictionary and Set.

1/3 Lists; are used in python to create lists of data in a sequential manner. They work from top to bottom, applying [] (known as the index) and typing as many strings inside as you want it will list in order from first to last, if you leave the brackets empty, it will produce no list. The list also arrays in order from 1 - (forever) depending on how long your list is, so you can name an element in your list e.g [jye, is, the, best] 'jye' would be 1, and 'best' would be 4.

2/3 Dictionary; is the way value data is saved into python. Dictionary may also be called one of these (maps, hashmaps, lookup tables, associative arrays). With the dictionary funcation in python, it gives the ability to search for a number or word in the directory (after inserting), this becomes very useful when storing lots and lots of data with different values to each, the best example would be how a phone book is structured, is generally the same layout as how a dictionary runs.

3/3 Set; is a data structure that gives you the ability to store and track any numeric number of values and choose between them in any set of order. Set is used by using the set() function that will then go on to using in an array. Sets cannot have the same elements (identical), if you do so it will get confused and error.

**** Q5. Describe the features of interpreters and compilers and how they are different.****

A compiler is a program on the computer that scans the entire program and transforms code written in a programming language into machine code.

It's a program that translates readable code to a language a computer can only understand, the language gets compiled into binary (1, 0) which humans cannot read.

A interpreter on the other hand is much faster in response, because instead of having to scan the whole program, it only scans a specific statement.

Q6.Identify TWO commonly used programming languages and explain the benefits and drawbacks of each.

Javascript

Javascript (JS) is text based programming tool in web developement that correlates with HTML and CSS. It was created in 1995 by Brendan Eich, and became a standard for use in 1997. JS is known to be used in client-side and server-side interactions. When a webpage has any moving objects, different types of graphics, interactive elements, its normally being done with JS.

Javascript allows users to use searchboxes, digital graphics that move around a webpage, zoom-in/out of photos, the use of the famous 'hamburger' menu, and much more. With all postives, there must come negatives, and they vary from debugging on client-side, to negative browswer support. But the main problem with JS is the client-side security, its often used for malicious purpuses, because the code is vieable via the person viewing it. It would be very easy to swap the code with something else and cause bad conduct.

Python

Python was created 1991, to give readability to programming. Since 1991 Python has gone through many stages of ups and downs, bringing it all the way from Python - Python3.0 almost 30 years later. Python gives a broad range of ability, vasting from over 15 data structures that all align with eachother. Python is a high-level programming language, It's known for scripting or better known as 'glue' for mixing components together.

One of the key features of Python is the error system. Because there's no compiling within the python tool, the debugging of the system reacts very quick with an answer on where you have got an error.(line 1, answer is a fish ><>) etc. This debugging tool becomes extremely useful when trying to figure out what you're doing wrong.

One of the downfalls of Python is its not a very fast language to type out, theres multiple structures that take optimal time to get correct without errors, and sometimes this works against you more than with you. Also there is no memory saving with Python, once you're out of a program/script it never saves the last memory, you are starting from scratch again, which in most cases is quiet an annoyying trait.

**** Q7. Identify TWO ethical issues from the areas below and discuss the extent to which an IT professional is ethically responsible in terms of the issue.**

Ethical issues are dependant on the person, everybody has different ethical values "moral principals that govern a persons behaviour or conducting of an activity". But with the law on the other hand, it doesnt value ethics as normal, it tries to be equal in every way possible. The two ethical issues i'll be going into more depth will be;

- 1. GPS tracking data and other types of metadata, MAC addresses, hardware fingerprints
- 2. aggressive sales and marketing practices designed to mislead and deceive consumers

1. The ethical issues surrounding the following are privacy and 2nd hand selling of tracking data/ meta data etc. "The Privacy Act 1988 (Privacy Act) is the principal piece of Australian legislation protecting the handling of personal information about individuals. This includes the collection, use, storage and disclosure of personal information in the federal public sector and in the private sector" As an IT professional I have the responsibility to produce complete privacy from one to another. that includes making sure all of the above is private.

This issue is also a part of the confidentiality agreement that as an IT professinal, I don't have the right to sell or make any of the information given or recieved to the public, or making the data available for purchase. By the rights of the law, I would only be able to talk about the data within my practice with people with authorization or given authorization.

2. The ethical issues surrounding agressive sales and marketing practices, mostly are a marketing scheme to get a customer to look or enter a store. I don't think these are ethically wrong, and it's not against the law to do so. An example of marketing practices designed to mislead and decieve consumers would be when a store has an advertisement online, saying "99% off" and when you get into store, theres only 1 item thats 99% off. This isn't against the law, its following the guidelines of (Misleading and selling guide)

As a professional I have the requirement to follow the law, and follow my ethical reasoning and there is countless problems with accepting any type of work with these actions in order. It would be ethically incorrect to put anybody in harms way for my own personal interest or profit.

Case Study Singtel Optus Pty Ltd (Optus) is a major supplier of telecommunications services in Australia. For approximately five months from Anzac Day 2010 Optus prosecuted a multi-media advertising campaign promoting its “Think Bigger” and “Supersonic” broadband data service plans to consumers. The campaign deployed advertisements in five different media: television, metropolitan and local newspapers, billboards, online and direct marketing. There were eleven separate advertisements and, although each advertisement in each medium was couched in different terms, each advertisement indicated in “headline” claims that the plans had an overall cap or quota, made up of distinct peak and off-peak quotas of broadband availability. Each headline claim was accompanied by a disclaimer in smaller and less prominent print noting “Speed limited once peak data exceeded”. In fact, once the peak quota was used up, the speed of the service was significantly slowed irrespective of the usage of the off-peak or overall quotas.

This case study file under the ethical issue of misealeading advertisement with the factor of promoting a "Think bigger, and Superosonic" broadband data plans that did not produce the correct information. As explained above in the ethical reasons, these principles have ben breached and as an IT professional I would have an ethical right to not work with a company that has done this to there loyal customers.

Q8. Explain control flow, using examples from the Python programming language

Control flow is a flow of progression, which transfers data from one to the other in a way that only helps the next line 'flow'. Control flow is used in python in the many different acts of the python language, from control statements to sequential statements, the flow of the way you have typed your code is the way python is going to understand your code, therefore its called the 'Control flow'. An example of control flow has been used in (Q13,14,15 and 16) but an example would be.

~~~
    ```
    i = 50
    x = 500
    o = 5000
    if i > x:
    if i > o:
        print("This is the flow")
    else:
        print("This is also flow")
    elif x > o:
        print("This might be control flow")
    else:
        print("CONTROL FLOW)
    ```
~~~

** Q9. Explain the difference between type coercion and type conversion. Are either of these used in Python?**

Both type coercion and type conversion do similar things. Its the way of converting a value from one data type to another data type (integer, string, float..). The difference between them is that type coercion is implicit (goes straight to the point, no vagueness) where as type conversion can be either explicit or implicit. Type conversion is used in python to swap the two data types around e.g (integers into strings) but type coercion isn't used in python.

** Q10 Explain data types, using examples **

Data types are the way python categorizes each type of data. There are 15+ date types that have all different abilities to the programming language 'python. Some of them are;

Strings also known as 'str' is one of the main data types (sequence) in the python programming language. They're normally surrounded by single or double quotation marks ('' and ""). Strings are a flow of text in characters. Strings are used when outputting a function with text, or to align a function with text.

Boolean also know as 'bool' is a key factor in python for its ability to give the truth of the value of an expression. Boolean data types array in only two functions, "True" and "False". Boolean is frequently used with other data types such as integers, floats, and strings.

Integars also known as 'int' are the numeric symbol for strings or other data types. Its a function thats needed when your producing a number. Int pretty much is a number without decimals, not negative, thats goes to an unlimited lenght. If it was to go to a number with decimals, it would become a float data structure. So in python, whenever asking for a number, you need to use 'int'.

** Q11 Here’s the problem: “There is a restaurant serving a variety of food. The customers want to be able to buy food of their choice. All the staff just quit, how can you build an app to replace them?” **

- Identify the classes you would use to solve the problem
- Write a short explanation of why you would use the classes you have identified

If I was to create a restuarant application, I would start from the top. A restuarant, you cannot make a buisness without one, and within the restuarant you will have four more classes. A chef, orders, a menu and staff. Without the restaurant you wouldn't need any of these, so that comes first.You need customers to order food for the chef so thats the starting point. You also are need a chef to cook the food, then staff to run food out, and run the restaurant.

restaurant { order[] chef[] menu[] staff[]
}

order { MenuItem Time_recieved delivery_method; (pickup or delivery) time for _pickup Time_till_arrival: customers_details: **(if ordering from inside restaurant)** getWhat_table getFood getPayment_options

}

chef { setChefs_name: getOrder_coming_in: getTime_of_order: getWhat_to_cook: setTime_of_completion
}

menu { Menu_options menu_prices description_of_food description_of_drinks

}

staff { accept_orders delivery transfer_food(around the restaurant) accept_payment }

** Q12 Identify and explain the error in the code snippet below that is preventing correct execution of the program **

```
# First of all there was no text in the input, so it was asking for
nothing in terminal. (This actually doesn't matter but it always helps.)
(It will still give an answer.)
# Converted celsius to a float, because when asking for a number in the
string, you need to convert with int or float.
```

```
celsius = float(input("Insert text here"))

fahrenheit = (celsius*9/5)+32

print(f"The result is: {fahrenheit}.")
```

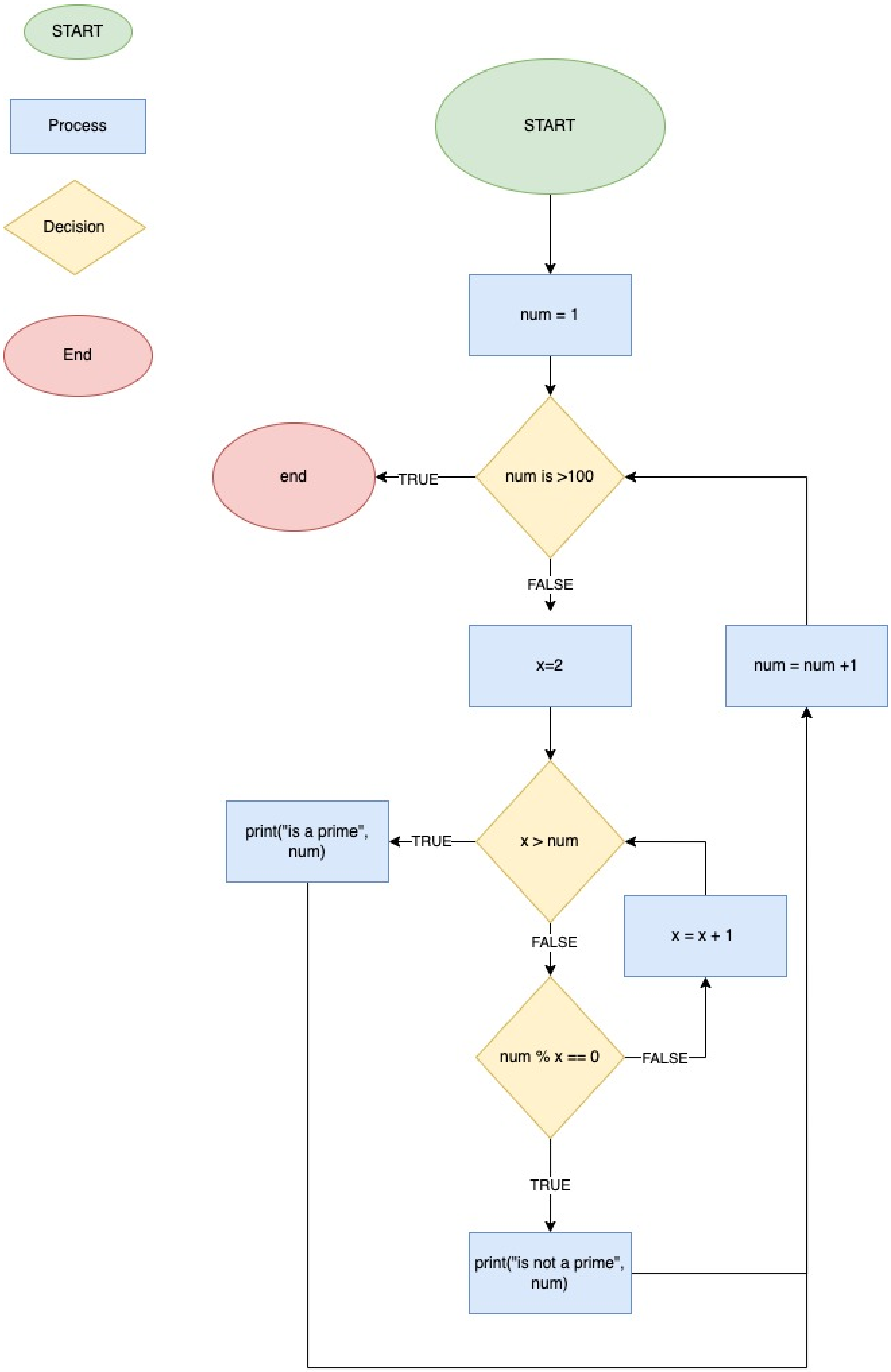
** Q13 The code snippet below looks for the first two elements that are out of order and swaps them; however, it is not producing the correct results. Rewrite the code so that it works correctly. **

```
    arr = [5, 22, 29, 39, 19, 51, 78, 97, 84]
    i = 0
    while (i < len(arr) -1):
        # and (arr[i] <= arr [i+1]
        if (arr[i] >= arr[i+1]):
            saved = arr[i+1]
            arr[i+1] = arr[i]
            arr[i] = saved
            print(arr)
            i = len(arr)

        i += 1
```

** Q14 Demonstrate your algorithmic thinking through completing the following two tasks, in order:**

1. Create a flowchart to outline the steps for listing all prime numbers between 1 and 100 (inclusive). Your flowchart should make use of standard conventions for flowcharts to indicate processes, tasks, actions, or operations
2. Write pseudocode for the process outlined in your flowchart


START
Process
Decision
End
START
num = 1
num is >100
TRUE
end
FALSE
x=2
num = num +1
x > num
TRUE
print("is a prime", num)
FALSE
x = x + 1
num % x == 0
FALSE
TRUE
print("is not a prime", num)

```
function is_prime(num)
    for x in range(2, num):
        if (num modulus x) == 0:
            return False
    return True

for i in range(1...100)
    if is_prime(i)
        print "{i} is prime"


# This is written in python

def is_prime(num):
    for x in range(2, num):
        if (num%x) == 0:
            return False
    return True

for i in range(1, 101):
    if is_prime(i):
        print("Is prime", i)
```

15 Write pseudocode OR Python code for the following problem:

You have access to two variables: raining (boolean) and temperature (integer). If it's raining and the temperature is less than 15 degrees, print to the screen "It's wet and cold", if it is less than 15 but not raining print "It's not raining but cold". If it's greater than or equal to 15 but not raining print "It's warm but not raining", and otherwise tell them "It's warm and raining"

```
raining = input("Is it raining?:")
temperature = int((input("How hot is it in degrees? ")))
if temperature < 15 and raining == "yes":
    print("it's wet and cold")
elif temperature < 15 and raining == "no":
    print("It's not raining but cold.")
elif temperature >= 15 and raining == "no":
    print("Its warm but no raining")
else:
    print("its warm and raining")
```

Q16 ACME Corporation are hiring a new junior developer, as part of their hiring criteria they've created a "coding skill score" based on the specific competencies they require for this role; the more important the skill is for ACME corp, the more points it contributes to the "coding skill score" The skills are weighted as follows:

- Python (1)
- Ruby (2)
- Bash (4)
- Git (8)
- HTML (16)
- TDD (32)
- CSS (64)
- JavaScript (128) Write a program that allows a user to input their skills and then tells them a) Their overall "coding skill score" b) Skills they may want to learn, and how much each one would improve their score

```
score = 0
result = ""

PYTHON = 1
RUBY = 2
BASH = 4
GIT = 8
HTML = 16
TDD = 32
CSS = 64
JAVASCRIPT = 128


skill = input("Do you know Python?:")

if skill == "yes":
    score = score + PYTHON
elif skill == "no":
    result = "Python (1 point)"

skill = input("Do you know Ruby?:")

if skill == "yes":
    score = score + RUBY
elif skill == "no":
    result = result + ",Ruby (2 points)"

skill = input("Do you know Bash?:")
if skill == "yes":
    score = score + BASH
elif skill == "no":
    result = result + ",BASH (4 points)"

skill = input("Do you know Git?:")
if skill == "yes":
    score = score + GIT
elif skill == "no":
    result = result + ",GIT (8 points)"

skill = input("Do you know HTML?:")
```

```
if skill == "yes":
    score = score + HTML
elif skill == "no":
    result = result + ",HTML (16 points)"

skill = input("Do you know TDD?:")
if skill == "yes":
    score = score + TDD
elif skill == "no":
    result = result + ",TDD (32 points)"

skill = input("Do you know CSS?:")
if skill == "yes":
    score = score + CSS
elif skill == "no":
    result = result + ",CSS (64 points)"

skill = input("Do you know Javascript??:")
if skill == "yes":
    score = score + JAVASCRIPT
elif skill == "no":
    result = result + ",JAVASCRIPT (128 points)"

print("Your score is:",score)
print("You should learn:", result)
```

This was another way I tried to do it, but kept failing at getting the points option added.

```

score = 0
result = ""
class skill:
    def __init__(self, name, value):
        self.name = name
        self.value = value

skills = [
    skill("PYTHON", 1),
    skill("RUBY", 2),
    skill("BASH", 4),
    skill("GIT", 8),
    skill("HTML", 16),
    skill("TDD", 32),
    skill("CSS", 64),
    skill("JAVASCRIPT", 128),
]

for skill in skills:
    knows_skill = input(f"Do you know {skill.name}?:")

    if knows_skill == "yes":
        score = score + skill.value
    elif knows_skill == "no":
        result = result + skill.name
        score2 = score

print("your score is: ",score)
print("You should learn:", result,"Total points lost:", score2)
```