

Q1

Research the development of the internet from 1980 to today. You must describe at least FIVE key events in the development of the internet. You can refer to events, people of significance, or technologies and how they have changed over time. 300 - 500 words

A1

Domain suffix

In 1985 the first domain suffix ".com" was claimed.

A domain suffix is the last part of a domain ".com". It can help identify the organisation the site is linked to e.g companies have .com, education institutions have .edu and education institutions based in australia will have edu.au.

This helps people identify both <https://www.sydney.edu.au/> and <https://www.uts.edu.au/> as australian education institutions and the remainder of the domain tell you that they are different australian education institutions

As webpages did not exist until 1991 domain suffix were mostly useful for emailing before that.

search engine

In 1990 Alan Emtage created the first search engine and called it Archie.

Search engines try to understand and categorise the content on a web page so that a user can provide keywords that the search engine can then provide ranked matches for.

Before search engines a user would need to know the site they wanted to go to.

('Search Engine History', 2006 - 2017)

World Wide Web

In 1991 the World Wide Web became available to the public. It was created by Tim Berners-Lee. Berners-Lee posted a summary of the project on the alt.hypertext newsgroup. This started the internet as we know it. It would fundamentally change the world.

The World Wide Web's technology was founded on what Berners-Lee did at CERN in the 80s with the European Organization for Nuclear Research. Berners-Lee was looking for a way to share information around the world without everyone that had access needing to be on the same hardware and software.

('20 years ago today', 2011)

WiFi

In 1997 wifi first released. The committee 802.11 was created. And from that IEEE802.11. This referred to a set of standards defining the communication for wireless. The basic specification for WiFi was created that allowed 2mb per second data transfers wirelessly between devices with the same WiFi specification. Then in 1999, WiFi was first introduced for personal home use.

('The History of WiFi', 2014)

Napster

On the 1st of June 1999 Napster was launched. It allowed people to share and access files without restriction. At its peak 57 million people were using Napster. On July 11, 2001, Napster shut down its entire network and has never been the same since.

Napster played a pivotal role bringing the openness and anonymous nature of the internet into the mainstream conversation.

(Napster, 2013)

Reference:

Search Engine History is published by Aaron Wall. 2006 - 2017

<http://www.searchenginehistory.com/>

20 years ago today, the World Wide Web was born - TNW Insider by Martin Bryant. 2011

<https://thenextweb.com/insider/2011/08/06/20-years-ago-today-the-world-wide-web-opened-to-the-public/>

The History of WiFi By Jessica Thomas. 2014

<https://purple.ai/blogs/history-wifi/>

Napster: the day the music was set free by Tom Lamont. 2013

<https://www.theguardian.com/music/2013/feb/24/napster-music-free-file-sharing>

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. 50 - 100 words per dot point

A2

Packets

A packet is the information that is being passed around on the internet or any computer network. Each packet contains a physical MAC address (media access control address (every network connectable device needs a mac address)), IP (Internet Protocol) addresses, port numbers, packet number and the data. Packets are the information being passed around a network. Nothing happens on any network including the internet without them.

IP (Internet Protocol) addresses

An unequal IP address is assigned to every device connected to the internet. If you didn't have a unequal address no packets could get sent to you.

IPV4 is a 128-bit address system with only 2 to the 32 available addresses.

2 to the 32 = 4,294,967,296

IPV6 is a 128-bit address system with only 2 to the 128 available addresses.

2to the 128 = 340,282,366,920,938,463,463,374,607,431,768,211,456

When it comes to IPV4 Most networks, like a home network only really have one device (a router) directly connected to the internet. This is not the case with IPV6. for better or for worse everything can be "directly" connected to the internet with IPV6

Routers and routing

A router redirects the network traffic as it has been set up. On a typical home network the router can be seen as the mediator between the internal and the routers internal network. The LAN (Local area network) is all the devices connected to each other that you have control over. The WAN (wider area network) is all the devices on the other side of your router. The WAN is normally the internet and you have very little control over it.

Routers and routing are an integral but of the internet. They are the web that connects all the devices and enables them to communicate with each other.

Routers do routing and have firewalls blocking unwanted traffic.

An example of routing is a web serving computer on a home network. As the web serving computer is not directly connected to the WAN, the home network's router would have port forwarding enabled, it would send ports 80 (HTTP) and 443 (HTTPS) or whatever was needed from the WAN to the web serving computer's LAN IP and visa versa.

DNS

DNS simply is a way of connecting a registered domain name (google.com) that people understand to an IP that a computer understands (142.250.66.174).

For people to be able to access the previously mentioned "web serving computer on a home network" There would need to be a registered domain linked with the WAN IP of the home network.

Someone enters the domain into a browser and thanks to DNS they are directed to the unequal IP.

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)

A3

TCP (Transmission Control Protocol)

TCP lets a device reliably send a packet to another device on the same network or on a different network. It is a direct connection between 2 devices. The information is not broadcast for all the devices on the network to deal with. Unlike UDP that broadcasts it to everything on the network. (The tcpip networking protocol suite)

HTTP (Hypertext Transfer Protocol)

Http is basically most of the content you see on a website. %99 of the time any text you see on a website it is written in HTTP. HTTP is the base language that websites will be built in.

HTTPS (Hypertext Transfer Protocol Secure)

HTTPS is HTTP with security. When you see a padlock icon on the left of the web address in your browser window you are using HTTPS. HTTPS creates an encrypted connection between your browser and the server you are connected to. Communication is securely being transferred between the browser and Server. You still need to trust the server but you know that information is getting to it without anything eavesdropping.

web browser

A web browser is an application installed on a device that a client uses to request information from a server. When the browser receives the information it can then render a webpage. Most of these webpages are on the world wide web but not necessarily. Modern browsers can also run their own applications called extensions. Modern browsers can also open many common files like pdf. Modern browsers can also have developer tools that let you inspect a webpage and edit the locally stored copy of it.

Reference:

The TCP/IP Networking Protocol Suite - dummies by dummies.com

<https://www.dummies.com/programming/networking/the-tcpip-networking-protocol-suite/>

Q4

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

50 - 100 words on each data structure

A4

Array

An array is a data structure that lists data and indexes each item starting from 0. For example,

```
trump_array = ["person", "woman", "man", "camera", "tv"]
```

"Person" is in array index 0 and "tv" is in array index 4.

The command `puts = trump_array[3]` would result in printing the string "camera".

binary tree

A binary tree is data structured where data is in a node. Each node also holds data as to whether it has a left and or right child node. Each node cannot have more than 2 child nodes. The first node is called the root. A node with no children is called a leaf.

If searching for a letter in an array of all the letters in the alphabet. The search would not be efficient as there are up to 26 searches made as it only returns (=) or (!=). If instead the alphabet letters were in a binary tree, the search can now get back (<), (=) or (>). It would get to the correct result much faster as the remaining items halve with each search.

hash table

A hash table is basically an Array but the index identifiers are replaced by a key.

This means in a hash you find something by its key and not its index.

If we had 2 arrays of personal details of 2 people.

```
person1 = ["Tom", "Smith", "20021985", "M"]
```

```
person2 = ["Jane", "Jones", "14081987", "F"]
```

To get one of the values you need to know the index.

If we used 2 hash tables the values now have keys pointing to them. The order doesn't matter.

```
person1 = { "firstname" => "Tom", "lastname" => "Smith", "dob" => "20021985", "sex" => "M" }
```

```
person2 = { "firstname" => "Jane", "dob" => "14081987", "sex" => "F", "lastname" => "Jones" }
```

Now to get the values we can just use the keys.

Q5

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

100 - 200 words on each way code is executed

A5

Compilers make a program while an interpreter runs the source code line by line.

The compiler will analyse the statements in the source code and check if it is correct. If the compiler thinks something is wrong it will return an error message. If no errors are found the compiler will create machine code from the source code. The compiler links all the different

source code files into one program that can be executed. In most cases a client is not going to want to use a terminal to run the application you have created for them.

Interpreters run the source code and all the linked source code files line by line. An Interpreter doesn't create an executable file and doesn't check everything before starting to running. In the life cycle of the development of an application, compilers are mostly only used when an executable file is needed or to check that an executable file can be made without a problem. The rest of the time developers will use Interpreters for testing. Interpreters make the development process much easier as the time it takes to start running tests is almost nothing. A little change can be made and tested very quickly.

Some programming languages like Python don't use a compiler at all. Python is an interpreted language and always runs line by line even in a shipped application.

Q6

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

200 - 400 words on each language

A6

C++

C++ was released in 1985 and was considered at the time to be a high-level programming language. Now it is mostly considered low-level as newer languages with all their readability and creature comforts have come out.

C++ is still a long way away from the machine code that it will be boiled down to for processing. But it is not as easy to understand at first glance and newer more high-level language like ruby, python, C# and many others.

Pros:

- Similar to other languages

C++ is similar to other languages in that if you were starting from scratch C++ would take you a long time to learn but if you already had experience in another common programming language your skill would be easily transferable.

- Wide support

C++ is used and loved by a lot of people. Online help files are accessible and easy to understand. C++ is active on forums like Stack Overflow and it will not take long to get a question you have answered.

- Small standard library

C++ has a relatively small standard library. Basically it gives a programmer less restrictions in what they can do.

- Speed

C++ is recognised as a very fast programming language most game where every millisecond counts are programmed in C++

Cons:

- Complex

C++ Programming is complicated and with a small standard library someone new to programming will struggle to learn it.

- Less memory management

A programmer will need for the most part do the memory management themselves as C++ does very little. Also adding to complex.

- Lack of custom operators

Many programming languages have custom operators. In C++ it is possible to redefine existing operators but creating a new operator is not possible.

- Lack of algebraic data types

C++ only has simple data types. A programmer will need to make their own or find a library if they want more.

- Syntactically strict

No flexibility in syntax. so difficult for a programmer to write code in a readable way.
(Pros and Cons of C++, 2018)

Python

Python was created in 1991 and is a high-level programming language. Python seems to be the programming language of choice for data analytics in the research space. This is because it is easy to get started with and easy to put raw data in and get raw data out. Many Enterprise Application are also created in Python.

Pros:

- Integration Feature

Python Integrates with Enterprise Application for easy development of Web services. Python also has powerful control capabilities for C, C++ or Java via Jython. Python runs on all modern operating systems.

- Expanded Support Libraries

Python has large standard libraries which include things like string operations, Internet, web service tools, operating system interfaces and protocols. Lots of the widely used programming tasks are already in Python. This helps limit the length of code in Python.

- Productivity

Python has a great option for building scalable multi-protocol network applications. This is possible because of Python's process integration features, unit testing framework and enhanced control capabilities. All together they help Python's productivity and speed for most applications.

Cons:

- Not adjustable to Other Languages

Python programs tend to become used to its features and libraries that they can struggle with working or learning other languages.

- Slow

Python runs with an interpreter and not a compiler, the step by step process of compilation and execution slow it down. Not everything needs to run very fast and Python will be fast enough for many applications.

- Weak in Mobile Computing

Python is good for many desktop and server platforms, but does not work well in programming for mobile use.

- Run-time Errors

Some Python developers say that it has many design restrictions. The final applications have errors popping up so even if it works fine more work needs to be done to get rid of the errors. (Pros and Cons of Python, 2018)

Reference:

Pros and Cons of C++ Programming Language by Pros-Cons, 2018

<https://www.pros-cons.net/pros-and-cons-of-c-programming-language/>

Pros and Cons of Python Programming Language by Pros-Cons, 2018

<https://www.pros-cons.net/pros-and-cons-of-python-programming-language/>

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.

List of topics containing ethical issues:

- access to a user's personal information (medical, family, financial, personal attributes such as sexuality, religion, or beliefs)
- intellectual property, copyright, and acknowledgement.
- criminal acts such as theft, fraud, trafficking and distribution of prohibited substances, terrorism
- GPS tracking data and other types of metadata, MAC addresses, hardware fingerprints
- freedom of thought, conscience, speech and the media
- aggressive sales and marketing practices designed to mislead and deceive consumers
- trading of shares on the stock exchange OR crypto-currencies

For each ethical issue identify a source of legal information relating to the ethical issue and discuss whether the law is helpful in assisting a developer to act in an ethical way. (Word count guide: 200 words max)

A7a

Access to a user's personal information

An IT professional has an ethical and legal obligation to the use of any user's personal information that they have access to only in the way that the user was informed their personal information would be used when it was collected.

A user agrees to what their data can be used for normally by checking an "I agree" checkbox on a large legal document they will never read. The ethical responsibility of any IT professional working with that personal information should be to minimise the divergence from the originally perceived use of their personal information by the user.

Privacy and Personal Information Protection Act 1998 No 133

(section 17) (Summarised)

A public sector agency holding personal information needs to make sure they are not using the information for something other than the purpose indicated to the individual when the information was collected.

Exceptions.

- The individual has consented to other purpose.
- The other purpose directly related to the purpose indicated to the individual.
- The other purpose is necessary in the prevention or minimization of serious and imminent threat to the life or health of the individual that the information is about.

Trading of shares on the stock exchange OR crypto-currencies

Insider Trading

An IT professional may have access to information before the general public. They have an ethical and legal obligation to not engage in Insider Trading. They also have an ethical responsibility to whistle blow when they see people not acting in the way the law allows.

Federal legislation - Corporations Act 2001

Volume 5 Chapter 7 Financial services and markets

Part 7.10 Market misconduct and other prohibited conduct relating to financial products and financial services

Division 3 The insider trading prohibitions

Definition of Inside information

Subdivision A—Preliminary

1042A Definitions (Summarised)

Inside information is information that is not generally available to a reasonable person.

Definition of Inside trader

Subdivision B The prohibited conduct

Section 1043A (Summarised)

A person (the insider) knows, or ought reasonably to know Inside information must not in any way apply for, acquire, or dispose of financial products. Also the insider must not, directly or indirectly, communicate the information, or cause the information to be communicated, to another person if the insider knows, or ought reasonably to know, that the other person would or would be likely to: Apply for, acquire, or dispose of financial products.

Reference:

Privacy and Personal Information Protection Act 1998 No 133

Current version 1 March 2020

<https://www.legislation.nsw.gov.au/view/html/inforce/current/act-1998-133#frnt-lt>

Federal legislation - Corporations Act 2001

Current version 28 September 2017

https://www.legislation.gov.au/Details/C2018C00424/Html/Volume_5#_Toc528572464

Conduct research into a case study of ONE of the ethical issues you have chosen discuss how an ethical IT professional should respond to the case study and how they might mitigate or prevent ethical breaches. (Word count guide: 400 - 600 words)

A7b

NewYork time Sept. 7, 2017

"Equifax, one of the three major consumer credit reporting agencies, said on Thursday that hackers had gained access to company data that potentially compromised sensitive information for 143 million American consumers, including Social Security numbers and driver's license numbers."

"On a scale of 1 to 10 in terms of risk to consumers, this is a 10," said Avivah Litan, a fraud analyst at Gartner."

(Equifax Says Cyberattack May... 2017)

In this case much more was stolen than the Social Security numbers of almost half of the population of the USA.

What are Social Security numbers

In 1936 the Social Security number was created by the Social Security Administration. It was created to track earnings of workers for Social Security benefits. Nowadays the Social Security number is used for many non Social Security things. Among other things it is used as an identity verification method for law enforcement, collecting child support payment and issuing credit. (Social Security Cards, 2005)

What could someone do if they have your Social Security number

In the USA someone can use a Social Security number to get other personal information about the owner of that Social Security number. This is identity theft and they can then get loans and credit cards that the owner of that Social Security number would not know about until the debt collectors start calling.

(Identity Theft and Your, 2017)

Identity theft

Almost 50% of the \$1.8 billion dollars lost by individuals in the USA in 2018 was because of Identity theft in the form of "Credit card fraud—new accounts".

(Facts + Statistics: Identity theft, 2019-2020)

What could have been done to stop this from happening

In this case it was a failure to update a webserver with a security patch that fixed a known flaw. Keeping security patches up to date on systems that have access to so data. A web server needs to be able to display unencrypted information from a server for a user to be able to see it. So I guess the lesson is do your security updates. Although it seems strange for one web server patch to be the only thing between hackers and 143 million Social Security numbers.

(U.S. Charges 4 Chinese... 2020)

IT security is important

It is tempting to cut corners or rely on other 3rd parties when it comes to your IT security due to tight deadlines and that fact that IT security is mostly invisible until it breaks. At every level of an organisation IT security needs to be taught about. During the development an application IT security should be thought about at every step. I think if this Equifax hack is going to be very useful for convincing them that post development support for IT security is a must.

Reference:

Equifax Says Cyberattack May Have Affected 143 Million in the U.S. By Tara Siegel Bernard, Tiffany Hsu, Nicole Perlroth and Ron Lieber, 2017

<https://www.nytimes.com/2017/09/07/business/equifax-cyberattack.html>

Social Security Cards: De Facto National Identification. By Jim Kouri, 2005

<https://web.archive.org/web/20120629234649/http://www.americanchronicle.com/articles/view/3911>

Identity Theft and Your Social Security Number. By The United States Social Security Administration, 2017

<https://www.ssa.gov/pubs/EN-05-10064.pdf>

Facts + Statistics: Identity theft and cybercrime. By Insurance Information Institute, 2019-2020

<https://www.iii.org/fact-statistic/facts-statistics-identity-theft-and-cybercrime>

U.S. Charges 4 Chinese Military Officers in 2017 Equifax Hack. By krebsonsecurity, 2020

<https://krebsonsecurity.com/tag/equifax-breach/>

Q8

Explain control flow, using an example from the Ruby programming language

100 words

A8

Flow control lets us control the flow of a program. Using a loop makes the program go around and around for as long as we want. Using IF statements directs the program down one path or another. Using cases lets us specify what path we want the program to take if a specified case is fulfilled.

An example in ruby is the simple if statement.

This program gets a number from 1 to 5 from the user.

We will use a "case" to point the flow of the program in 6 different directions and get 6 different results.

```
print "please enter a number from 1 to 5 : "  
user_input = gets.tr('^1-5', '').to_i  
case user_input  
when 1  
  puts "Yellow"  
when 2  
  puts "Lime"  
when 3  
  puts "Pink"  
when 4  
  puts "Amber"  
when 5  
  puts "The quick brown fox jumps over the lazy dog"  
else  
  puts "invalid input"  
end
```

Reference:

Ruby flow control by Jan Bodnar, 2020

<http://zetcode.com/lang/rubytutorial/flowcontrol/>

Q9

Explain type coercion

100 words

A9

Type coercion is the changing of data into another type of data (see next question).

It is common to need to change a user's input from the default string to a number. We would need type coercion in order to do comparisons with known data.

Let's say we ask a user for their date of birth in order to verify them. They enter "14101980" which is a string. If the DOB we have in the database is an integer then we need to type coercion the users input string into an integer before any comparisons can be made.

Q10

Explain data types, using examples

100 words

A10

Data types are the allowable operations of values.

Booleans - true or false

Symbols - a lighter form of strings. They are preceded by a colon (:)

Numbers - The main two types of numbers used are Integer and floats. An Integer is a whole number and a float needs to have a decimal point in it. Take the integer 24 and turn it into a float and you get 24.0

Strings - a string is made up of multiple characters. They are differentiated by being placed inside quotes.

Arrays - an array can store multiple data items of all types. Items in an array are separated by a comma.

Hashes - a hash is an array but stores key-value pairs instead. A key-value pair has a key and then a value e.g {"banana" => \$1.5}

Reference:

Data types in Ruby by educative, Undated

<https://www.educative.io/edpresso/data-types-in-ruby>

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

N/A words

A11

In this situation the MVP would be likely to be.

I have been given the word "food" but will use "dish" instead.

Menu management

Page where the restaurant management could

- Add a dish to the menu.
- Edit availability
- The dish would need a name, Cost, availability, description, vegetarian status, gluten free status, vegan status and halal status.
- View order history

Menu for customers

- List of all available dishes with its cost and description ect..
- Shopping cart.
- Payment facilities.
- Customer gets a booking number after ordering.

Kitchen

- Print out of orders.

Objects needed

- Menu
- Dishes
- Shopping carts
- Orders

Classes

- Dishes
- Shopping carts

I feel like only Dishes and the Shopping cart should be classes because of their complexity.

The Menu is a simple coalition of available dish attributes.

The Orders are a simple list generated from a complicated Shopping cart.

The Dishes are complex as they have lots of attributes that need to be set at the creation of the dish. Also we would want class functions like "Update availability" and "update price"

The Shopping carts are complex because they need to be created with no dishes. They also need to update their cost and check dish availability with the server regularly. We would probably want additional class functions like "generate order" and all the functions needed to pay for shopping carts.

Q12

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

100 words

```
celsius = gets
fahrenheit = (celsius * 9 / 5) + 32
print "The result is: "
print fahrenheit
puts "."
```

A12

```
`#User input (gets) needs to be trimmed (.tr) down to remove any non numerical data including
full stops ('^0-9.', '') as we only want a number.
#User input (gets.tr) now needs to be turned into a float. Mathematical calculation will be broken
by having a string anywhere in them.
celsius = gets.tr('^0-9.', '').to_f
fahrenheit = (celsius * 9 / 5) + 32
print "The result is: "
#This calculation works however there should not be a full stop at the end of a float. Changing
"print fahrenheit" to "puts fahrenheit" and removing "puts "."
puts 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.

N/A words

```
arr = [5, 22, 29, 39, 19, 51, 78, 96, 84]
i = 0
while (i < arr.size - 1 and arr[i] < arr[i + 1])
  i = i + 1 end
puts i
arr[i] = arr[i + 1]
arr[i + 1] = arr[i]
```

A13

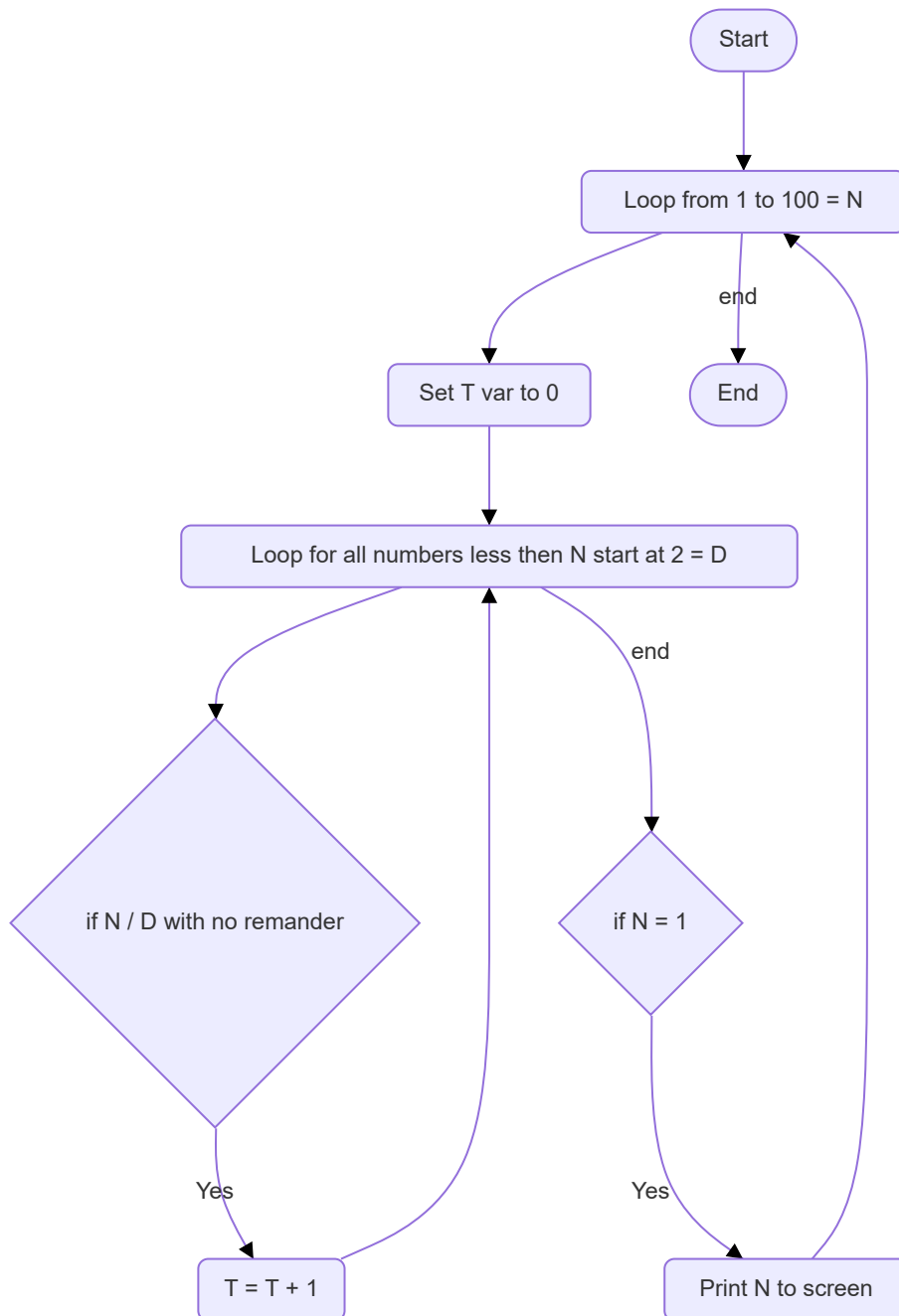
```
#look for the first two elements that are out of order and swaps them
arr = [5, 22, 29, 39, 19, 51, 78, 96, 84]
i = 0
while (i < arr.size - 1 and arr[i] < arr[i + 1])
  i = i + 1
end
#puts i ... don't need this puts
#added a memory var (m) for value at position i
m = arr[i]
arr[i] = arr[i + 1]
#replace arr[i + 1] with memory var (m)
arr[i + 1] = m
puts arr
```

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

#flowchart using mermaid-js



Pseudocode

```
#Loop though numbers 1 to 100 (number)
#create set tester var set to 0
#for each number get all the numbers less then it -1 from 2 up. (divider) eg. we don't can want 1
or the number
#if number is divisible by divider with 0 remainder then get a tester score
#after the devider loop is all finished if a numbers tester score is 1 it is only devisable by itself and
1 because we start at 2. Print it
```

Code

```

for i in (1..100)
  t = 0
  for divider in (2..i)
    if i % divider == 0
      t = t + 1
    end
  end
  if t == 1
    puts i
  end
end

```

Q15

Write pseudocode OR Ruby 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".

N/A words

A15

```

if raining = true and temperature < 15
  puts "It's wet and cold"
elsif raining = false and temperature < 15
  puts "It's not raining but cold"
elsif raining = false and temperature >= 15
  puts "It's warm but not raining"
else
  puts "It's warm and raining"
end

```

Q16

An allergy test produces a single numeric score which contains the information about all the allergies the person has (that they were tested for). The list of items (and their value) that were tested are:

- eggs (1)
- peanuts (2)
- shellfish (4)
- strawberries (8)
- tomatoes (16)
- chocolate (32)
- pollen (64)
- cats (128)

So if Tom is allergic to peanuts and chocolate, he gets a score of 34.

Write a program that, given a person's score can tell them:

- a) whether or not they're allergic to a given item
- b) the full list of allergies.

N/A words

A14

solution description

remove largest possible allergen value from score until score == 0 each time a value is removed
record allergen type

pseudocode

Create allergic_to array

Create a Var=128

Create allergen list array

Ask User for their score

convert users_score to int

Create loop that runs 8 times

find highest allergen types score that is > or = users_score using another loop

Add removed allergen type to allergic_to

take away allergen types score from users_score

If after 8 loops users_score is not = to 0 then tell user invalid score

Ask user if they want

A: Select an allergen from the list and see if you are allergic to it

or

B: List all allergies

or

C: Quit

Terminal App

```
allergic_to = []
num = 128
allergies = ["cats", "pollen", "chocolate", "tomatoes", "strawberries",
"shellfish", "peanuts", "eggs"]
puts `clear`
print "Enter your allergie score: "
users_score = gets.tr('^0-9', '').to_i
for i in (0..7)
  if users_score >= num
    allergic_to << i
    users_score = users_score - num
  end
  num = num / 2
end
if users_score != 0
  puts "invalid score"
  exit
end
while true
  puts `clear`
  puts "How would you like the results?"
```



```

puts "A: Select an allergie from the list and see if you are allergic to it"
puts "or"
puts "B: List all allergies"
puts "or"
puts "C: Quit"
print "Please enter A, B or C: "
user_view = gets.chomp.upcase
if user_view == "A"
  puts `clear`
  go = 1
  while go == 1
    puts ""
    puts "1.cats    2.pollen    3.chocolate    4.tomatoes    5.strawberries
6.shellfish    7.peanuts    8.eggs"
    print "Enter the number next to the corresponding allergie or '9' to
go back: "
    user_select_num = (gets.tr('^0-9', '').to_i-1)
    if user_select_num == 8
      go = 0
    else
      len = (allergic_to.length()-1)
      hit = 0
      for i in (0..len)
        if user_select_num == allergic_to[i]
          hit = 1
          puts `clear`
          print "Yes you are allergic to "
          puts allergies[user_select_num]
        end
      end
      if hit == 0
        puts `clear`
        print "No you are not allergic to "
        puts allergies[user_select_num]
      end
    end
  end
end
elsif user_view == "B"
  puts `clear`
  print "You are allergic to: "
  len = (allergic_to.length()-1)
  for i in (0..len)
    print allergies[allergic_to[i]]
    if i != len
      print ", "
    end
  end
  puts "."
  exit
elsif user_view == "C"
  exit
else
  puts "Invalid selection"
end
end

```

