Foundations of Computing

Tutorial/Workshop ○ ○ ○ ○

Week 12

Today's Tutorial

○   ○   ○   ○

1 Bases, conversion, encoding

2 HTML

3 ACM

Numeric bases

Decimal system – base 10

Binary system – base 2

Octal system – base 8

Hexadecimal system – base 16

Numeric bases

Binary is good for computers because they store data in bytes (0 or 1)

Hexadecimal also good because can store 4 bits in 1 hex character

Decimal is not a great number base to work with; must convert, which leads to rounding errors

Unicode

Graphic character symbol		Hexadecimal character value													
	0020	0 0030	@ 0040	P 0050	` 0060	p 0070	00A0	° 00B0	À 00C0	Ð 00D0	à 00E0	ð 00F0
! 0021	1 0031	A 0041	Q 0051	a 0061	q 0071	¡ 00A1	± 00B1	Á 00C1	Ñ 00D1	á 00E1	ñ 00F1
" 0022	2 0032	B 0042	R 0052	b 0062	r 0072	¢ 00A2	² 00B2	Â 00C2	Ò 00D2	â 00E2	ò 00F2
# 0023	3 0033	C 0043	S 0053	c 0063	s 0073	£ 00A3	³ 00B3	Ã 00C3	Ó 00D3	ã 00E3	ó 00F3
$ 0024	4 0034	D 0044	T 0054	d 0064	t 0074	¤ 00A4	´ 00B4	Ä 00C4	Ô 00D4	ä 00E4	ô 00F4
% 0025	5 0035	E 0045	U 0055	e 0065	u 0075	¥ 00A5	µ 00B5	Å 00C5	Õ 00D5	å 00E5	õ 00F5
& 0026	6 0036	F 0046	V 0056	f 0066	v 0076	¦ 00A6	¶ 00B6	Æ 00C6	Ö 00D6	æ 00E6	ö 00F6
' 0027	7 0037	G 0047	W 0057	g 0067	w 0077	§ 00A7	· 00B7	Ç 00C7	× 00D7	ç 00E7	÷ 00F7
(0028	8 0038	H 0048	X 0058	h 0068	x 0078	¨ 00A8	¸ 00B8	È 00C8	Ø 00D8	è 00E8	ø 00F8
) 0029	9 0039	I 0049	Y 0059	i 0069	y 0079	© 00A9	¹ 00B9	É 00C9	Ù 00D9	é 00E9	ù 00F9
* 002A	: 003A	J 004A	Z 005A	j 006A	z 007A	ª 00AA	º 00BA	Ê 00CA	Ú 00DA	ê 00EA	ú 00FA
+ 002B	; 003B	K 004B	[005B	k 006B	{ 007B	« 00AB	» 00BB	Ë 00CB	Û 00DB	ë 00EB	û 00FB
, 002C	< 003C	L 004C	\ 005C	l 006C	| 007C	¬ 00AC	¼ 00BC	Ì 00CC	Ü 00DC	ì 00EC	ü 00FC
- 002D	= 003D	M 004D	] 005D	m 006D	} 007D	00AD	½ 00BD	Í 00CD	Ý 00DD	í 00ED	ý 00FD
. 002E	> 003E	N 004E	^ 005E	n 006E	~ 007E	® 00AE	¾ 00BE	Î 00CE	Þ 00DE	î 00EE	þ 00FE
/ 002F	? 003F	O 004F	_ 005F	o 006F	007F	¯ 00AF	¿ 00BF	Ï 00CF	ß 00DF	ï 00EF	ÿ 00FF

Unicode

The standard of text encoding

Assigns every text character from every language to a value

Gives a universal encoding for the whole world

Documents can now have multiple languages in them

Others – subsets of unicode

ASCII
Allows 128 different characters, enough for English but not all languages

UTF-32
Each character stored in 32 bit unit, takes a lot of space

UTF-8
Each character stored in 8–32 bits

UTF-16
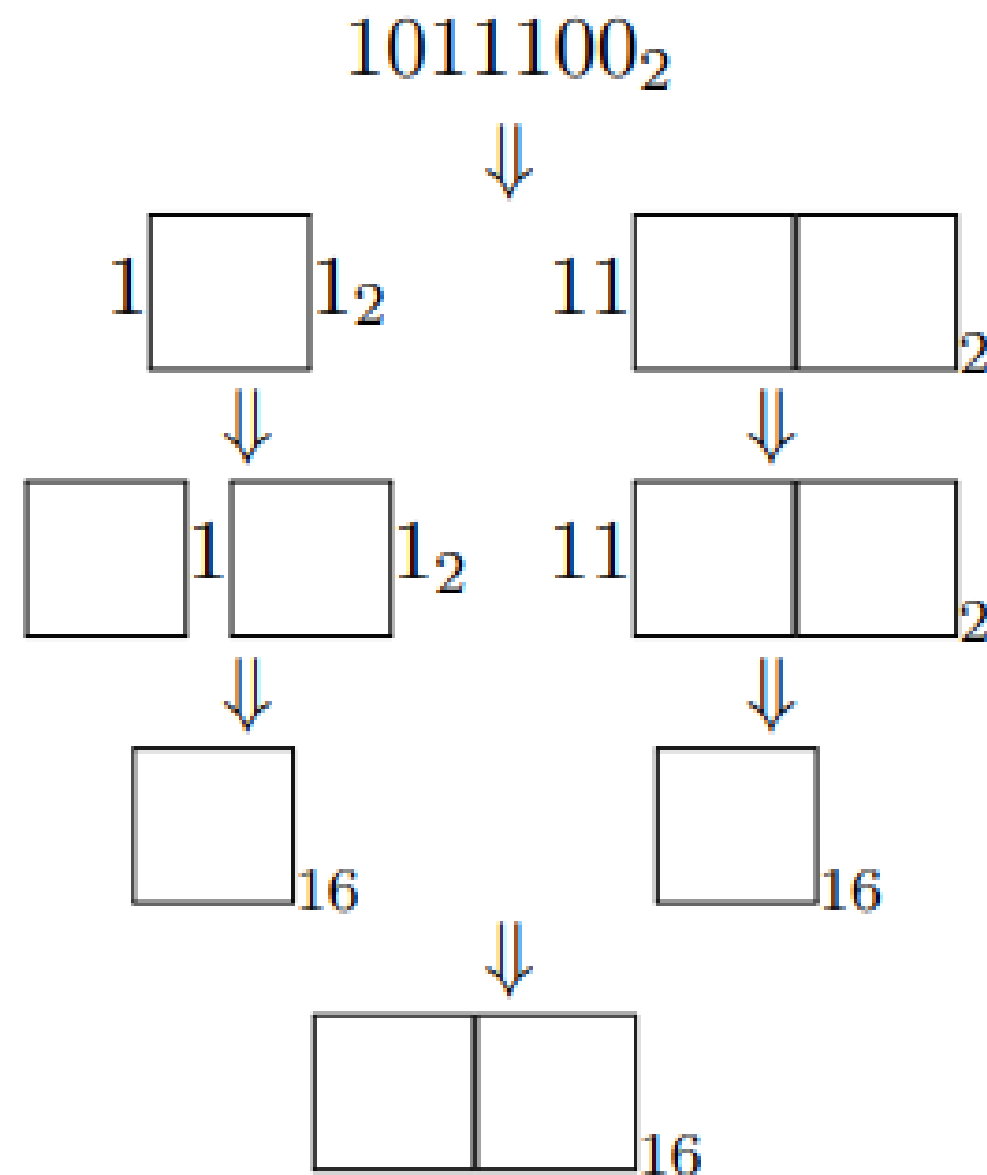Each character stored in 16–32 bits

Allow more variation

Exercise 1

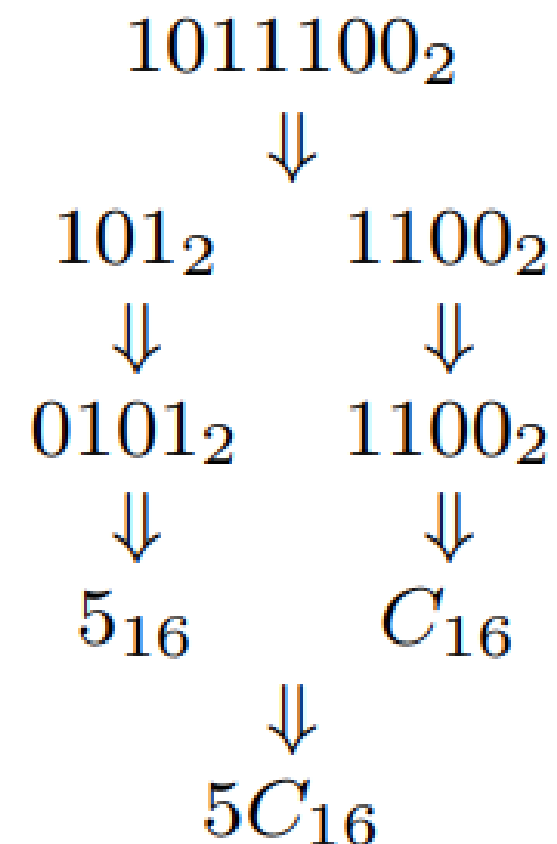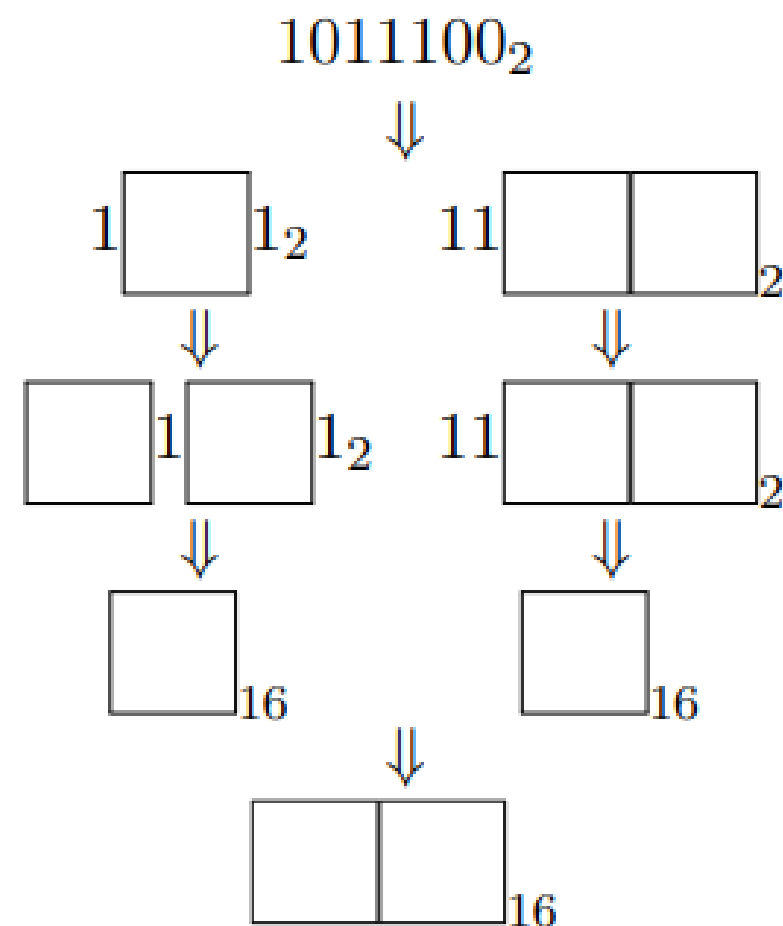# Exercise 1

(a) Convert the binary number 1011100 to hexadecimal by filling in the boxes in the following diagram with a single digit:

$$1011100_2$$

Decimal	Binary	Hexa	Decimal	Binary	Hexa
0	0000	0	8	1000	8
1	0001	1	9	1001	9
2	0010	2	10	1010	A
3	0011	3	11	1011	B
4	0100	4	12	1100	C
5	0101	5	13	1101	D
6	0110	6	14	1110	E
7	0111	7	15	1111	F

Exercise 1 answer

1011100_2

$\Downarrow$

1 ☐ 1_2 11 ☐☐ $_2$

$\Downarrow$ $\Downarrow$

☐ 1 ☐ 1_2 11 ☐☐ $_2$

$\Downarrow$ $\Downarrow$

☐ $_{16}$ ☐ $_{16}$

$\Downarrow$

☐☐ $_{16}$

1011100_2

$\Downarrow$

101_2 1100_2

$\Downarrow$ $\Downarrow$

0101_2 1100_2

$\Downarrow$ $\Downarrow$

5_{16} C_{16}

$\Downarrow$

$5C_{16}$

Step 1: separate into 4-bit sequences
Step 2: Add leading zeroes to make them all four bits long
Step 3: Directly convert binary numbers (0000-1111) into hexadecimal numbers (0-F)
Step 4: Combine hexadecimal numbers together, retaining place value of the original binary sequence
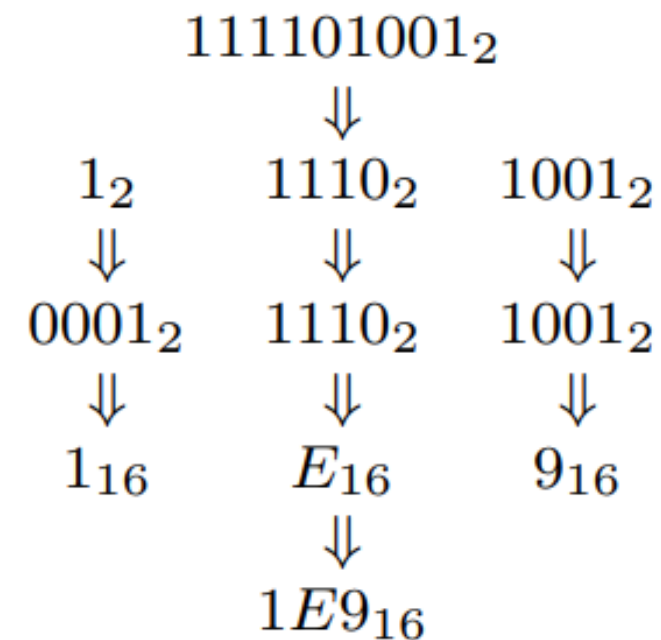
Exercise 1

(b) Convert the binary number 111101001 into hexadecimal using a method like the one shown above.

Decimal	Binary	Hexa	Decimal	Binary	Hexa
0	0000	0	8	1000	8
1	0001	1	9	1001	9
2	0010	2	10	1010	A
3	0011	3	11	1011	B
4	0100	4	12	1100	C
5	0101	5	13	1101	D
6	0110	6	14	1110	E
7	0111	7	15	1111	F

Exercise 1 answer

(b) Convert the binary number 111101001 into hexadecimal using a method like the one shown above.

A:

$$111101001_2$$
$$\Downarrow$$

$$1_2 \qquad 1110_2 \qquad 1001_2$$
$$\Downarrow \qquad \Downarrow \qquad \Downarrow$$
$$0001_2 \qquad 1110_2 \qquad 1001_2$$
$$\Downarrow \qquad \Downarrow \qquad \Downarrow$$
$$1_{16} \qquad E_{16} \qquad 9_{16}$$
$$\Downarrow$$
$$1E9_{16}$$

Step 1: separate into 4-bit sequences
Step 2: Add leading zeroes to make them all four bits long
Step 3: Directly convert binary numbers (0000-1111) into hexadecimal numbers (0-F)
Step 4: Combine hexadecimal numbers together, retaining place value of the original binary sequence

URL

Uniform Resource Locater
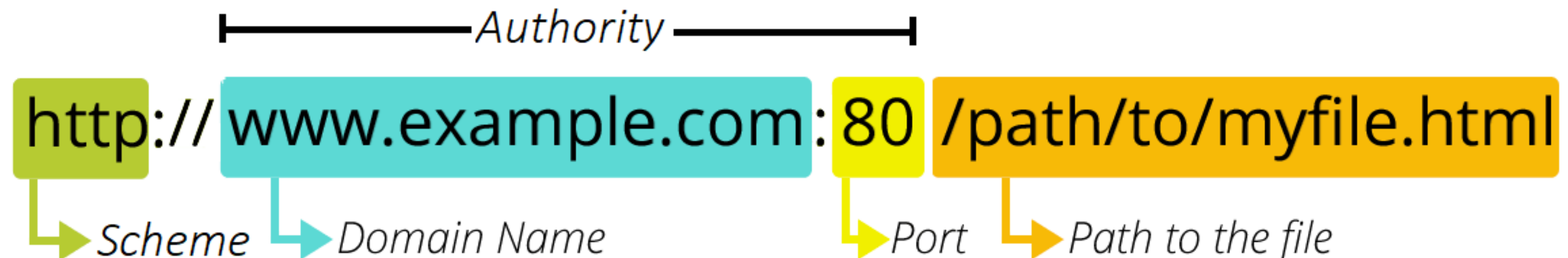
Identifies the "path" to a particular file or resource, accessed via a "scheme", located at a "host" communicated to via a "port"

HTML

Hypertext Markup Language

(not programming!)

Used exclusively on the web, to describe how a webpage should be displayed

e.g. colours, fonts, text, columns, images, headings

HTML tags – structural

```
<html> declares it is an HTML document
<head> header
<body> document body

<html>
  <head>
  </head>
  <body>
  </body>
</html>
```

HTML tags – text formatting

\<b\> bold \</b\>

\<i\> italic \</i\>

<u>\<u\> underline \</u\></u>

\<h1\> Heading 1 \</h1\>

\<h2\> Heading 2 \<h2\>

\<h3\> Heading 3 \</h3\>

\<h4\> Heading 4 \</h4\>

HTML tags – lists and tables

```
<ol>
<li>Apple</li>
<li>Mango</li>
<li>Grapes</li>
<li>Pineapple</li>
<li>Orange</li>
</ol>
```

1. Apple
2. Mango
3. Grapes
4. Pineapple
5. Orange

`<ul>` unordered list

`<ol>` ordered list

`<li>` a list item

```
<table>
  <tr>
    <td>cell 1</td>
    <td>cell 2</td>
    <td>cell 3</td>
  </tr>

  <tr>
    <td>cell 4</td>
    <td>cell 5</td>
    <td>cell 6</td>
  </tr>
</table>
```
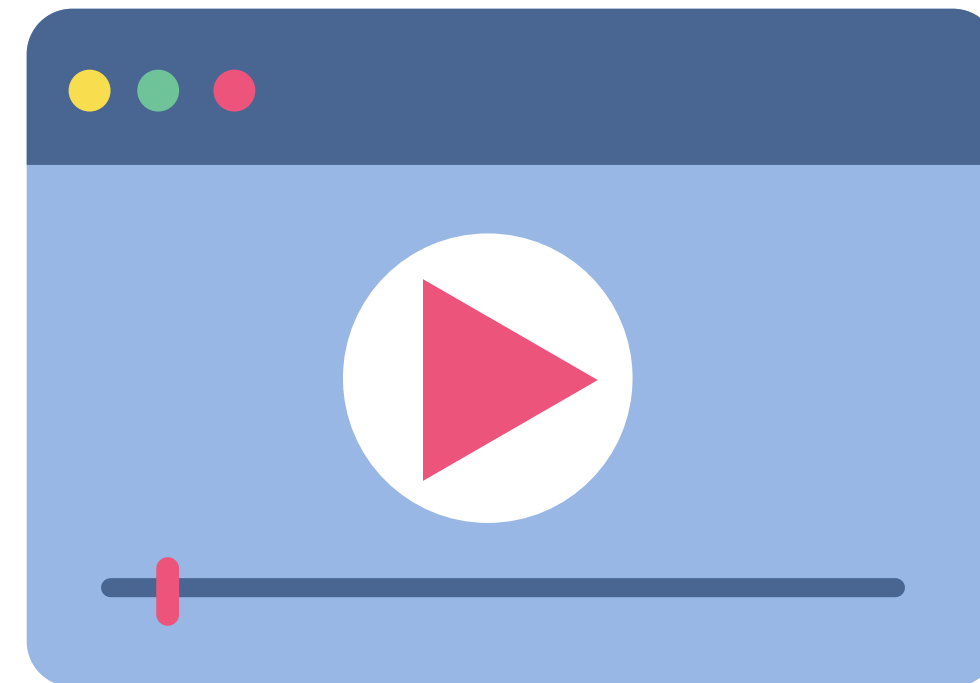
HTML TABLES

cell 1	cell 2	cell 3
cell 4	cell 5	cell 6

`<table>` table

`<tr>` table row

`<td>` table cells

HTML tags – media

`<a  href="www.google.com"> This is google </a>`   This is google

`<img src="apple.png">`

`<video width="320" height="240" controls>`
  `<source src="movie.mp4" type="video/mp4">`
`</video>`

HTML entity

A special character

e.g. < and > are used in html, so cannot be written on their own

use < and > instead

Static vs dynamic HTML

Static page looks the same to each user

e.g. Wikipedia page

Dynamic page looks different

e.g. Grok, we will see our own name and code for each

question

Exercises 2&3

Exercise 2 answer

Exercise 3

```
context = {"name": "Dragon",
           "desc": "A_legendary_creature._Breathes_fire!",
           "power": "100",
           "picture_file": "https://upload.wikimedia.org/wikipedia/commons"
                           "/thumb/6/6b/Easy_origami_dragon_for_beginners-"
                           "_how_to_paint_a_dragon_for_beginners.jpg/320px"
                           "-Easy_origami_dragon_for_beginners-_how_to_pai"
                           "nt_a_dragon_for_beginners.jpg"}
customise_html("template.html", context, "out.html")
```

```html
<body>
  <div class="container-fluid">
    <div class="card" style="width: 18rem;">
      <img src="{{ picture_file }}" class="card-img-top" alt="{{ name }}">
```

nary. For example, if the template contained the snippet {{ shape }},
with context = {"shape": "triangle"}, then the output file would contain triangle.

Exercise 3 answer

A:

```python
from html import escape
TOKEN_LEN = 2
def customise_html(template_file, context, outfile):
    with open(template_file, 'r') as file, open(outfile, 'w') as out:
        template = list(file.read()); idx = 0
        while idx < len(template):
            if template[idx:idx + TOKEN_LEN] == ["{", "{"]:
                start = idx + TOKEN_LEN
                end = start + template[idx + TOKEN_LEN:].index("}")
                key = "".join(template[start:end]).strip()
                template[idx:end + TOKEN_LEN] = escape(context.get(key, ""))
            idx += 1
        out.write("".join(template))
```

ACM Code of Ethics and Professional Conduct

1. Contribute to society and to human well-being, acknowledging that all people are stakeholders in computing

2. Avoid harm

3. Be honest and trustworthy

4. Be fair and take action not to discriminate 5. Respect the work required to produce new ideas, inventions, creative works, and computing artefacts

6. Respect privacy

7. Honour confidentiality

Dual use

Technologies which can be equally used for **good** and **malicious** purposes

Many implementations of artificial intelligence and automation can be classified as dual use e.g. ChatGPT, self-driving cars

Options post–FOC

1. **Stop here!**
2. **Foundations of Algorithms (COMP10002)**
   a. Challenging
3. **Elements of Data Processing (COMP20008)**
   a. Python
   b. Data analysis
4. **Database Systems (INFO20003)**
   a. Well–taught
   b. Interesting, but not helpful outside computing

WORKSHOP

o   o   o   o

Grok, problems from sheet, ask me questions :)