

Full Name	George Wilkinson
Roll Number	00677611
ID Number	CHC119
E-Mail	<a href="mailto:G.Wilkinson2@edu.salford.ac.uk">G.Wilkinson2@edu.salford.ac.uk</a>

## CSI Task 1

**a) Choose a number between 51 and 70. This will be number A. Choose another number between 80 and 120. This will be number B. Using 2's complement 8-bit binary arithmetic, calculate number A – number B giving your answer in binary and decimal. (5 marks)**

Number A - 64 - 0100 0000

Number B - 96 - 0110 0000

Binary Subtraction	
Num A	0100 0000
Num B	0110 0000
Flip B	1001 1111
Add 1	1010 0000
Num A	0100 0000
Num B	1010 0000
Output	1110 0000
1s Comp	0001 1111
2s Comp	-0010 0000
Decimal	-32

**b) Choose one number from the set { 30, 31, 33, 34, 35, 35, 37, 38, 39 } this is numberC. Choose a number from { 3, 5, 6 } this is numberD. Using 8-bit binary arithmetic, calculate numberC × numberD giving your answer in binary and decimal. (5 marks)**

Number C - 34 - 0010 0010

Number D - 6 - 0000 0110 -  $2^1$ ,  $2^2$

Binary Multiplication	
Num C	0010 0010
Mult $2^1$	0100 0100
Num C	0010 0010
Mult $2^2$	1000 1000
Add	1100 1100
Decimal	204

Here, we split our exponent into 2, giving a value of 2 and 4 ( $2^2$ ), so we can multiply the mantissa by both, and add them together.

**c) Using numberC and numberD from (b), use binary arithmetic to calculate: numberC + numberD giving your answer in binary and decimal. Convert numberC into IEEE-754 format. (5 marks)**

<b>numberC</b>	<b>0010 0010</b>
Floating Point	0010 0010 . 001
C	100010.001
Decimal	34.125
<b>numberD</b>	<b>0000 0010</b>
Floating Point	0000 0110 . 011
D	110.011
Decimal	6.375
<b>Binary Addition</b>	
Num C	0010 0010.001
Num D	0000 0110.011
Add	0010 1000.100
Decimal	40.5

Here, we add the floating points to the two numbers, and convert both to decimal to check our answer. Adding them together in the 3rd table to get an answer in both decimal and binary.

Convert -> Floating Point	Binary	Decimal
Num C	100010.001	34.125
Normalised	$1.00010001 \times 2^5$	34.125
Remove leading 1	0.00010001	0.06640625
Sign	0 (Positive)	0
Exponent	10000100	$132 (2^5)$
Mantissa	000100010000000000000000	1.06640625
Total	01000010000010001000000000000000	$2^5 \times 1.06640625$

Here, since the sign is positive, it takes the value of 0. The exponent is  $2^5$ , but in IEEE-754, you would take the exponent from the normalised value and add to the bias (127 in 32bit). I then got the mantissa by removing the leading 1, since it is assumed, from the normalised value, and adding the trailing 0's to get 23 bits. Concatenating these values I then get the final answer in decimal and binary.

**d) Choose a number between 140 and 160 - this is numberE. Convert numberE to hexadecimal. Choose a number between 170 and 255 - this is numberF. Convert numberF into octal. Using 8-bit binary, calculate numberE OR numberF. Give your answer in binary, octal, decimal and hexadecimal. (5 marks)**

NumberE - 160

NumberF - 223

NumberE	160
Binary	0b10100000
Hexadecimal	0xA0

NumberF	223
/8	27r7
/8	3r3
/8	0r3
Octal	0337
Binary	1101 1111

NumE OR NumF	
NumE	0b1010 0000
NumF	0b1101 1111
OR (Binary)	0b11111111
OR (Decimal)	255
OR (Hexadecimal)	0xFF
OR (Octal)	0377

Here, I have picked two numbers, the first of which is converted to hexadecimal, and the second converted to octal using a division conversion method on the decimal value. Then I have done a bitwise operation for OR on the two binary values to get a binary value, and then converted to decimal, hexadecimal and octal - in this case it is the maximum value for each using 8 bit binary.

## CSI Task 2

**a) Choose a short phrase of between 50-60 characters. You could choose a website headline, song title, etc. Convert this phrase using Caesar shift encryption and a key of 5 to create your ciphertext. Carry out a cryptanalytic attack to try to work out the decryption key and plaintext. (10 marks)**

F1 veteran to retire and take up new role in 2024.

Original String	F1 veteran to retire and take up new role in 2024.
Key = 1	G1 wfufsbo up sfujsf boe ublf vq ofx spmf jo 2024.
Key = 2	H1 xgvgtcp vq tgvtkg cpf vcmg wr pgy tqng kp 2024.
Key = 3	I1 yhwhudq wr uhwluh dqg wdnh xs qhz uroh lq 2024.
Key = 4	J1 zixiver xs vixmvi erh xeo i yt ria vspi mr 2024.
Key = 5	K1 ajywjfs yt wjynwj fsi yfpj zu sjb wtqj ns 2024.

Character	Count
j	7

Character	Count
w	4
s	4
f	3

The most common English letter is E, so I am going to substitute J for E, since I assume it will be the most frequent character.

E <- F <- G <- H <- I <- J

This results in a key of 5, so translating the 2nd word: ajyjwtfs -> veteran. This looks promising as it resulted in an English word. Trying the 4th word: wjynwj -> retire. Since this word is also in English, I can assume this candidate key is likely to work, so I will translate the entire string.

K1 ajyjwtfs yt wjynwj fsi yfpj zu sjb wtqj ns 2024. -> F1 veteran to retire and take up new role in 2024.

**\*\*b)** Computer A sends 5 packets of data to computer B using Sliding Windows Flow Control. The transmission time (time to put on the network) for a packet of data is 1 'time units'. Transmission time for an acknowledgement is 0 'time units' (they are very small). The propagation time (time to travel through network) for any transmission is random (between 3 and 5 'time units', you choose a random time for each packet and acknowledgement sent). B's packet processing time is 2. B cannot process multiple packets simultaneously. The initial window size is 2. Draw a diagram to show how flow will be controlled while the data is being sent. (10 marks)

Transmission Time = 1tu

Acknowledgement = 0tu

Propagation Time = 3-5tu

Processing Time (B) = 2tu

Window Size = 2

For propagation time, I am going to use the following values:

( 0 = 3, 1 = 3, 2 = 4, 3 = 4, 4 = 5), and for simplicity's sake, I will mirror the propagation time there with the propagation time back for the acknowledgement.

Time	A	B	C	D	E	F	G	H	I	Sender	In Transit →	In Transit ←	Recipient
1		0								Sending	0		Idle
2		0	1							Sending	0,1		Idle
3		0	1							Blocked	0,1		Idle
4		0	1							Blocked	1		Process 0
5		0	1							Blocked			Process 0, send ACK1
6		0	1							Blocked		ACK1	Process 1
7		0	1							Blocked		ACK1	Process 1, send ACK2
8		0	1							Blocked		ACK1, ACK2	Idle
9	0		1	2						Sending	2	ACK2	Idle
10	0		1	2						Blocked	2	ACK2	Idle
11	0		1	2	3					Sending	2,3		Idle
12	0		1	2	3					Blocked	2,3		Idle
13	0		1	2	3					Blocked	3		Process 2
14	0		1	2	3					Blocked	3		Process 2, send ACK3
15	0		1	2	3					Blocked		ACK3	Process 3
16	0		1	2	3					Blocked		ACK3	Process 3, send ACK4
17	0		1	2	3					Blocked		ACK3, ACK4	Idle
18	0		1	2	3					Blocked		ACK3, ACK4	Idle
19	0		1	2	3	4				Sending	4	ACK4	Idle
20	0		1	2	3	4				Idle	4	ACK4	Idle
21	0		1	2	3	4				Idle	4		Idle
22	0		1	2	3	4				Idle	4		Idle
23	0		1	2	3	4				Idle	4		Idle
24	0		1	2	3	4				Idle			Process 4
25	0		1	2	3	4				Idle			Process 4, send ACK5
26	0		1	2	3	4				Idle		ACK5	Idle
27	0		1	2	3	4				Idle		ACK5	Idle
28	0		1	2	3	4				Idle		ACK5	Idle
29	0		1	2	3	4				Idle		ACK5	Idle
30	0		1	2	3	4				Idle		ACK5	Idle
31	0		1	2	3	4				Idle			Idle

## Linux Task 1

- Task 1.1 (6 marks): Create the following hierarchy of directories/subdirectories and run the tree command to check that all is correct and add images of the commands and tree/find output.

```
boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & Linux/Coursework 1$ mkdir Test\ 2023
boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & Linux/Coursework 1$ cd Test\ 2023/
boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & Linux/Coursework 1/Test 2023$ mkdir CSIL PDP
boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & Linux/Coursework 1/Test 2023$ mkdir CSIL/Linux PDP/CV
boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & Linux/Coursework 1/Test 2023$ mkdir CSIL/Linux/Workshop\ 1
boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & Linux/Coursework 1/Test 2023$ tree
```

```
.
├── CSIL
│   ├── Linux
│   │   └── Workshop 1
└── PDP
    └── CV
```

5 directories, 0 files

- Task 1.2 (3 marks): Use vi to make a file called "grep.txt" inside the "CV" directory. The content of the file should include your full name. Use nano to make a file called "cv.txt" inside the "Linux" directory. The content of the file should include your favourite module. Use cat to output the contents of your files to the screen and add an image of this to your assignment. Run the tree command to show that your files are in the

correct location and add and image of the commands used and the output of tree/find to your assignment output.

```
boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & Linux/Coursework 1/Test 2023$ vi PDP/CV/grep.txt
boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & Linux/Coursework 1/Test 2023$ nano CSIL/Linux/cv.txt
boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & Linux/Coursework 1/Test 2023$ cat PDP/CV/grep.txt
George Wilkinson
boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & Linux/Coursework 1/Test 2023$ cat CSIL/Linux/cv.txt
Computer Systems Internals and Linux!!!
boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & Linux/Coursework 1/Test 2023$ tree
.
├── CSIL
│   ├── Linux
│   │   ├── cv.txt
│   │   └── Workshop 1
│   └── PDP
│       └── CV
│           └── grep.txt
└── 5 directories, 2 files
```

- Task 1.3 (3 marks): Using absolute filenames (ones that start with a /), move the grep.txt file into the “Workshop 1” directory. cd into the “PDP” directory. Using relative filenames (.. notation ones that don't start with a /), copy the cv.txt file into the “CV” directory. cd into the “CSIL” directory. Using relative filenames ones that don't start with a /), rename the “Workshop 1” directory so it is called “Tutorial 1” Run the tree command and include images of the commands and the tree/find output. Your final output should also include your command history.

```
boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & Linux/Coursework 1/Test 2023$ mv /home/boris/OneDrive/Computer\ Science\ Year\ 1/Computer\ Systems\ Internals\ \&\ Linux/Coursework\ 1/Test\ 2023/PDP/CV/grep.txt /home/boris/OneDrive/Computer\ Science\ Year\ 1/Computer\ Systems\ Internals\ \&\ Linux/Coursework\ 1/Test\ 2023/CSIL/Linux/Workshop\ 1/
boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & Linux/Coursework 1/Test 2023$ tree
.
├── CSIL
│   ├── Linux
│   │   ├── cv.txt
│   │   ├── Workshop 1
│   │   │   └── grep.txt
│   └── PDP
│       └── CV
└── 5 directories, 2 files
boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & Linux/Coursework 1/Test 2023$ cd PDP
boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & Linux/Coursework 1/Test 2023/PDP$ cp ../CSIL/Linux/cv.txt CV/
boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & Linux/Coursework 1/Test 2023/PDP$ cd ../CSIL/
boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & Linux/Coursework 1/Test 2023/CSIL$ mv Linux/Workshop\ 1 Linux/Tutorial\ 1
```



boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & L  
inux/Coursework 1/Test 2023\$ tree

```
.
├── CSIL
│   ├── Linux
│   │   ├── cv.txt
│   │   └── Tutorial 1
│   │       └── grep.txt
│   └── PDP
│       ├── CV
│       └── cv.txt
```

5 directories, 3 files

## Linux Task 2 (grep)

- Find the “famiclone” consoles?

```
inux/Coursework 1$ cat consoles.csv | grep -E "*famiclone*"
Dendy famiclone,Home,Micro Genius,1992,6
Pegasus famiclone,Home,Micro Genius,1991,1
```

- Find all lines with words with 4 consecutive consonants.

```
ux/Coursework 1$ cat consoles.csv | grep -E "[b-df-hj-np-tv-zB-DF-HJ-NP-TV-Z]{4}"
Nintendo Switch,Hybrid,Nintendo,2017,129
Master System,Home,Sega,1986,12
Master System Brazil,Home,Tectoy,1989,8
Famicom Disk System,Home console add-on,Nintendo,1986,4
Magnavox Odyssey,Home,Magnavox Philips,1978,2
Atari Lynx,Handheld,Atari,1989,1
```

- How many Hybrid consoles have been released?

```
boris@thinkpad:~/OneDrive/Computer Science Year 1/Computer Systems Internals & L  
inux/Coursework 1$ cat consoles.csv | grep -E "*Hybrid*" | wc -l
1
```

- Find all consoles with one word names.

```
ux/Coursework 1$ cat consoles.csv | grep -E "^[^,[:space:]]+,"
Platform,Type,Firm,Released,Units sold millions
PlayStation,Home,Sony,1994,102
Wii,Home,Nintendo,2006,101
Xbox,Home,Microsoft,2001,24
GameCube,Home,Nintendo,2001,21
Dreamcast,Home,Sega,1998,9
WonderSwan,Handheld,Bandai,1999,3
Intellivision,Home,Mattel,1980,3
N-Gage,Handheld,Nokia,2003,3
ColecoVision,Home,Coleco,1982,2
Telstar,Dedicated,Coleco,1976,1
```

- Find consoles whose name is at least 25 characters.

```
inux/Coursework 1$ cat consoles.csv | grep -E "^[^,]{25,}"
Game Boy and Game Boy Color,Handheld,Nintendo,1989,118
Super NES Classic Edition,Dedicated,Nintendo,2017,5
3DO Interactive Multiplayer,Home,The 3DO Company,1993,2
```

- What console names end in a number?

```
Year 1/Computer Systems Internals & Linux/Coursework 1$ cat consoles.csv | grep
-E "^[^,]+[0-9],"
PlayStation 2,Home,Sony,2000,155
PlayStation 4,Home,Sony,2013,117
PlayStation 3,Home,Sony,2006,87
Xbox 360,Home,Microsoft,2005,84
PlayStation 5,Home,Sony,2020,41
Nintendo 64,Home,Nintendo,1996,32
Atari 2600,Home,Atari,1977,30
Quest 2,VR headset,Reality Labs Meta,2020,20
PC Engine TurboGrafx-16,Home,NEC Hudson Soft,1987,10
Sega SG-1000,Home,Sega,1983,2
Atari 7800,Home,Atari,1986,1
Atari 5200,Home,Atari,1982,1
```

- Find the consoles that have sold 100 million or more.

```
linux/Coursework 1$ cat consoles.csv | grep -E '([^|,)([1-9][0-9]{2,})($|,)'
PlayStation 2,Home,Sony,2000,155
Nintendo DS,Handheld,Nintendo,2004,154
Nintendo Switch,Hybrid,Nintendo,2017,129
Game Boy and Game Boy Color,Handheld,Nintendo,1989,118
PlayStation 4,Home,Sony,2013,117
PlayStation,Home,Sony,1994,102
Wii,Home,Nintendo,2006,101
```

- What year had the most consoles released?

```
linux/Coursework 1$ cat consoles.csv | grep -Eo '([^|,)([1-9][0-9]{3})|$\n' | sort |
uniq -c | sort | tail -n 1
4 ,1990
```

## Linux Task 3 (sed)

3.1: Change all occurrences of 'Sony' to 'Better than XBox'.

```
ux/Coursework 1$ cat consoles.csv | sed -r "s/Sony/Better than XBox/g"
Platform,Type,Firm,Released,Units sold millions
PlayStation 2,Home,Better than XBox,2000,155
Nintendo DS,Handheld,Nintendo,2004,154
Nintendo Switch,Hybrid,Nintendo,2017,129
Game Boy and Game Boy Color,Handheld,Nintendo,1989,118
PlayStation 4,Home,Better than XBox,2013,117
PlayStation,Home,Better than XBox,1994,102
Wii,Home,Nintendo,2006,101
PlayStation 3,Home,Better than XBox,2006,87
Xbox 360,Home,Microsoft,2005,84
```

3.2: Change all commas “,” to colons “:”.

```
/Computer Systems Internals & Linux/Coursework 1$ cat consoles.csv | sed -r "s/,/:/g"
Platform:Type:Firm:Released:Units sold millions
PlayStation 2:Home:Sony:2000:155
Nintendo DS:Handheld:Nintendo:2004:154
Nintendo Switch:Hybrid:Nintendo:2017:129
Game Boy and Game Boy Color:Handheld:Nintendo:1989:118
PlayStation 4:Home:Sony:2013:117
PlayStation:Home:Sony:1994:102
Wii:Home:Nintendo:2006:101
PlayStation 3:Home:Sony:2006:87
Xbox 360:Home:Microsoft:2005:84
```

3.3: Change all consoles years from the 20th century (19xx) to “antique”.



```

coursework 1$ cat consoles.csv | sed -r "s/\b(19[0-9]{2})\b/Antique/g"
Platform,Type,Firm,Released,Units sold millions
PlayStation 2,Home,Sony,2000,155
Nintendo DS,Handheld,Nintendo,2004,154
Nintendo Switch,Hybrid,Nintendo,2017,129
Game Boy and Game Boy Color,Handheld,Nintendo,Antique,118
PlayStation 4,Home,Sony,2013,117
PlayStation,Home,Sony,Antique,102
Wii,Home,Nintendo,2006,101
PlayStation 3,Home,Sony,2006,87
Xbox 360,Home,Microsoft,2005,84

```

3.2: Change all commas "," to colons

```

/Computer Systems Internals 6 L
Platform:Type:Firm:Released:Uni
PlayStation 2:Home:Sony:2000:15
Nintendo DS:Handheld:Nintendo:2
Nintendo Switch:Hybrid:Nintendo
Game Boy and Game Boy Color:Har
PlayStation 4:Home:Sony:2013:11
PlayStation:Home:Sony:1994:102
Wii:Home:Nintendo:2006:101
PlayStation 3:Home:Sony:2006:87
Xbox 360:Home:Microsoft:2005:84

```

3.4: With lines that contain "Hybrid" append the line "Runs better when plugged into a TV."

```

coursework 1$ cat consoles.csv | sed -r "/Hybrid/ s/$/ Runs better when plugged into a TV./"
Platform,Type,Firm,Released,Units sold millions
PlayStation 2,Home,Sony,2000,155
Nintendo DS,Handheld,Nintendo,2004,154
Nintendo Switch,Hybrid,Nintendo,2017,129 Runs better when plugged into a TV.
Game Boy and Game Boy Color,Handheld,Nintendo,1989,118
PlayStation 4,Home,Sony,2013,117
PlayStation,Home,Sony,1994,102
Wii,Home,Nintendo,2006,101
PlayStation 3,Home,Sony,2006,87
Xbox 360,Home,Microsoft,2005,84

```

```

PlayStation 2,Home,Sony,2000,155
PlayStation,Home,Sony,1994,102
Wii,Home,Nintendo,2006,101
PlayStation 3,Home,Sony,2006,87
Xbox 360,Home,Microsoft,2005,84

```

3.5: Change all commas "," to colons

```

/Computer Systems Internals 6 L
Platform:Type:Firm:Released:Uni
PlayStation 2:Home:Sony:2000:15
Nintendo DS:Handheld:Nintendo:2
Nintendo Switch:Hybrid:Nintendo
Game Boy and Game Boy Color:Han
PlayStation 4:Home:Sony:2013:11

```