#Sheet 6
Arnav
ICS

6.1

a.

Simplicity carried to the extreme, becomes elegance.
- Jon Franklin

Work

I did it on paper but the general idea is for the first number 53, first convert to decimal so 5*16+ 3*16^0= 83, 83 now if we look in the ascii table it is letter s. And so on for other numbers. 20 corresponds to a space, 2c is comma, 2e- fullstop. 0a is newline. 6e end.

b.

UTF-8	Unicode	Name
7c	U+007C	Vertical Line
7b	U+007B	Left Curly Braces
e2 88 92	U+2212	Minus
7d	U+007D	Right curly bracket
e2 88 aa	U+222A	Union
7b	U+007B	Left Curly Braces
C2 ad	U+00AD	SOFT HYPHEN
7d	U+007D	Right curly bracket
e2 88 aa	U+222A	Union
7b	U+007B	Left Curly Braces
e2 80 93	U+2013	En Dash
7d	U+007D	Right curly bracket
e2 88 aa	U+222A	Union
e2 80 91	U+2011	Non breaking hyphen
7d	U+007D	Right curly bracket
e2 88 aa	U+222A	Union
7b	U+007B	Left Curly Braces
e2 80 94	U+2014	Em Dash
7d	U+007D	Right curly bracket
e2 88 aa	U+222A	Union

7b	U+007B	Left Curly Braces
e2 80 92	U+2012	Figure dash
7d	U+007D	Right curly bracket
7c	U+007C	Vertical Line
20	0X20	Space
3d	U+003D	Equal sign
20	0X20	Space
37	U+0037	Digit seven
0a	U+000A	Line Feed

c. In UTF 32 use 5 bytes for a character therefore 8000000*5= 40000000 bytes
For UTF 8 uses 4 bytes therefore 8000000*3= 24000000 bytes

6.2

a. 1 and 3 are not equivalent since even if we subtract 2:00hrs (offset) the time will be the same but not the day.

2 , 4, 6 are equivalent since adding 2:00 hrs to the time will make 4 2019-10-13T17:15:00 which is equivalent to 2.Also 6 is equivalent to 4 and hence to 2 as well because if we subtract 12hrs we will go back one day and in the evening 17:15 therefore 6 becomes 2019-10-13T17:15:00

3 and 5 are equivalent since if we add 12:45 to 5 then we will get 13:15 in the afternoon which is equivalent to time of 3. Note the day doesn't changed since we started at midnight.

b. If the time in UTC is known, but the offset to local time is unknown, this can be represented with an offset of "-00:00".

c. This problem arises because most C programs use a library of routines called the standard time library .Aa signed 4-byte integer has a maximum value of 2,147,483,647, and this is where the Year 2038 problem comes from. The maximum value of time before it rolls over to a negative (and invalid) value is 2,147,483,647, which translates into January 19, 2038. On this date, any C programs that use the standard time library will start to have problems with date calculation.
To fix it can be resolved by switching to a 64-bit system.