Introduction to Computer Systems

Department of Computer Science and Information Systems

Lecturer: Steve Maybank

sjmaybank@dcs.bbk.ac.uk

Spring 2020

Revision of the Summer 2017 Examination

Question 1a

- Add the decimal integers 15 and 27
- Show your working

	2	7
+	1, carry	5
Answer:	4	2

Question 1b

- Subtract the binary number 101 from the binary number 11001.
- Show your working.

	1	1	borrow, 0	0	1
-		repay	1	0	1
Answer:	1	0	1	0	0

Question 1c

- Write out the binary representation of
  $$2^6 + 2^4$$
- 2^6 is 1000000
- 2^4 is 10000
- Adding: $1000000 + 10000 = 1010000$

Question 1d

- Explain how the binary representation of a number specifies that number as a sum of powers of 2

- The ith place in the representation, reading right to left is associated with 2^{i-1}.

- The number is the sum of powers $2^{i-1}$for which the digit in the ith place is 1

Question 2a

- Which of the following are Boolean statements?

- (3+7)*2
- (3+7) == 2
- C = 4
- 6 < 10

- Answer: those that have a value True or False

Question 2b

- Write out the truth table for A OR B.

A	B	A OR B
0	0	0
1	0	1
0	1	1
1	1	1

- Truth tables for NOT A, A AND B, A XOR B?

Question 2c

- Write out a Boolean expression that is true if x is strictly less than y or strictly greater than y+5

- $x < y$ OR $x > y+5$

Question 3a

- Obtain the Brookshear floating point representation of 3+(1/4)

- Recall $\pm 2^r * 0.t$
- Recall sign bit, exponent and mantissa
- +11.01

Question 3b

- What feature allows representations of very large and very small numbers near to 0?
- Recall $\pm 2^{r} * 0.t$

Question 4a

- Explain the terms track and sector for a hard drive
- Why do the tracks have their shape?

Question 4b

- A hard drive has a capacity of 4 TB. The data rate for reading is 100 MB per s.
- How many seconds are required for reading the whole disk?

- 0.1 GB per s
- 1 GB in 10 s
- 1 TB = 1000 GB
- 4 TB = 4000 GB read in 4000*10 = 40000 s

Question 5a

- Give an example of a two-dimensional array of integers.
- Answer: a 3x3 array:

4	1	0
0	4	0
2	3	2

Question 5b

- How is it possible to store a 2 dimensional array in a one dimensional memory?
- Answer: one row at a time:

4	1	0
0	4	0
2	3	2

4	1	0	0	4	0	2	3	2

Question 5c

- Write a pseudo code algorithm to add the numbers in a one dimensional array A and print the result.

- Other tasks: max, min, find i such that

  $A[i] < A[i+1]$

Question 5c Continued

```
sum = 0
i = 0
while i < length(A)
    sum = sum+A[i]
    i = i+1
endWhile
print(sum)
```

Question 6a

- Explain the action of the instruction with op code 8

Answer: 8RST, bitwise And the contents of registers S and T. Put the result in register R. E.g.

S	0	1	1	0	0	1	1	1
T	0	0	1	1	1	1	1	0
R	0	0	1	0	0	1	1	0

Question 6b

- Write a program to load the contents of memory cell 91 into a register, set the rightmost 4 bits to 0 and store the resulting bit string in cell 92.

- Answer:

1191	Load register 1 with the bit pattern in cell 91
22F0	Load register 2 with F0 = 11110000
8312	And registers 1 and 2, put result in register 3
3392	Store the register 3 bit pattern in cell 92

Question 7a

- Define the term algorithm. Why are algorithms important?

- Answer (bookwork): an ordered set of unambiguous executable steps that defines a terminating process.
- An algorithm is required for any task to be performed by a computer.

Question 7b

- Why is there no algorithm for printing all the integers less than 5?
- Answer: the process requires an infinite number of steps

Question 7c

- Implement a loop in a program. Write out a pseudo code example.

- Answer: in a while loop a Boolean expression is evaluated. If the expression is true, then a block of code is executed and the expression is evaluated again. If false, then the code following the while loop is executed

Question 7c (Example)

- Pseudo code example of a while loop

```
i = 0
while i < 5
    print(i)
    i = i+1
endWhile
```

Question 8a

- Describe one advantage and one disadvantage of a linked list.

- Answer (bookwork): the different items in the list can be stored anywhere in memory

- To access an element it is necessary to search the list item by item

Question 8b

- Describe the way in which the head pointer and the null pointer are used.
- Answer (bookwork): the value of the head pointer is the location of the first element on the list.
- The null pointer marks the end of the list.

Question 8c

- Replace the item B in the list with the item D

addresses	10	11	12	13	14	15	16	17	18	19
contents	H	12	A	16	C	0	B	14	D	0

initial	H	12	→	A	16	→	B	14	→	C	0
updated	H	12	→	A	18	→	D	14	→	C	0

Question 9a

- Why is a sequential file appropriate for storing music?
- Answer: when the music is played the records in the file are accessed in the same order that they are stored in the file. This makes access efficient.

Question 9b

- Describe the structure of an index file.
- Answer (bookwork): The data in the file is stored in a list of records. Each record is identified by a unique key. The file contains an index which consists of pairs (k, a) where k is the key of a record and a is the address of the location where the record is stored.

Question 9c

- An index file contains at most 2^6 records. The file is stored in a memory with 2^{14} cells. What is the maximum size of the index in bits?

- Answer: Each pair (k, a) requires 6 bits for the key and 14 bits for the address. There are at most 2^6 pairs. The maximum size of the index is

  $2^6 \times (6+14) = 1280$ bits

Question 10a

```
function gcd(m, n)
          while (m ≠ n)
                    r = m-n
                    m = maximum(n, r)
                    n = minimum(n, r)
          endWhile
          return m
endFunction
```

- What happens if gcd is called with $m > 0$ and $n = 0$?

Question 10a (Continued)

```
function gcd1(m, n)
        if n == 0, return m
        endIf
        while (m ≠ n)
                r = m-n
                m = maximum(n, r)
                n = minimum(n, r)
        endWhile
        return m
endFunction
```

- Write out a new function gcd1 that returns the GCD if $m \geq n > 0$ and returns m if $m > n = 0$.

Question 10b

```
function gcd2(m1, n1)
        if m1 ≥ n1
                return gcd(m1, n1)
        else
                return gcd(n1, m1)
        endIf
endFunction
```

- Write out a new function gcd2 that returns the GCD if $m \geq n > 0$ or if $n > m > 0$.