Exercise 7: Basics of computer science, part II (Solution)

Problem 1: Binary Arithmetic

1. Find the 8-bit two's complement representation of the following integers: -23, -42.

Answer:

$23 \xrightarrow[\text{8 bit}]{\text{binary}} 00010111 \xrightarrow{\text{invert}} 11101000 \xrightarrow{\text{add 1}} 11101001 = -23_{10}$

$42 \xrightarrow[\text{8 bit}]{\text{binary}} 00101010 \xrightarrow{\text{invert}} 11010101 \xrightarrow{\text{add 1}} 11010110 = -42_{10}$

2. Solve using two's complement: $-13 + 52 = ?$.

Answer:

$13 \xrightarrow[\text{8 bit}]{\text{binary}} 00001101 \xrightarrow{\text{invert}} 11110010 \xrightarrow{\text{add 1}} 11110011 = -13_{10}$

$52 \xrightarrow[\text{8 bit}]{\text{binary}} 00110100$

Adding both:

$$\begin{array}{cccccccccc} & 1 & 1 & 1 & 1 & & & & & \\ & & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 \\ + & & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ \hline & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \end{array}$$

Ignoring the MSB (Most Significant Bit), we get 00100111_2.
$100111_2 = 39_{10}$, which is exactly what we expect.

3. Solve the following expressions in binary representation: $126 \cdot 16 = ?$, $\frac{74}{2} = ?$, $\frac{91}{8} = ?$

Answer:

When multiplying a base-10 number by 10, we simply add a 0 to the number: $157 \times 10 = 1570$. When dividing by 10, we simply move the period to the left: $\frac{634.17}{10} = 63.417$. For 100 we perform the same operations only twice ($100 = 10^2$), for 1000 three times ($1000 = 10^3$), etc. The same applies for powers of 2 in binary! Since $2 = 2^1, 8 = 2^3, 16 = 2^4$, we can easily convert to binary, move the period the right places, and convert back to decimal:
$126_{10} = 1111110_2$, multiplying by $16 = 2^4$ yields 11111100000 (adding four 0s). Converting back to decimal yields 2016_{10}.

Same for 74: $74_{10} = 1001010_2 \xrightarrow{\text{move period 2 places left}} 100101.0 \xrightarrow{\text{back to dec}} 37_{10}$.

Same for 91: $91_{10} = 1011011_2 \xrightarrow{\text{move period 3 places left}} 1011.011 \xrightarrow{\text{back to dec}} 11.375_{10}$

Problem 2: Floating Point

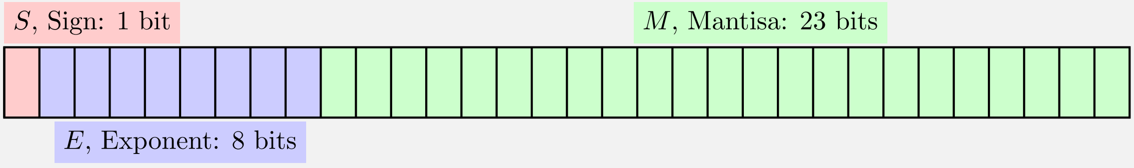
1. How many bits are in a standard floating point number? In a double-precision floating point number?

Answer:

A standard (IEEE-754) floating point number has 32 bits. A double-precision floating point number has 64 bits.

2. Name the different components of a floating point number.

Answer:



3. Convert the following 32-bit floating point number to a real base-10 number:
   1 1000 0001 011 0000 0000 0000 0000 0000.

Answer:

1	1	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

$S = 1$, $E = 1000\ 0001$, $M = 011\ 0000\ 0000\ 0000\ 0000\ 0000$.

(a) $S = 1$ means this is a negative number.

(b) The exponent E must is relative to -127. In this case $E = 1000\ 0001_2 = 129_{10}$, and so the total exponent is $E = 129 - 127 = 2$.

(c) The mantisa is normalized, and so $M = 1.011\ 0000\ \ldots$, which means $M = 1 \times 2^0 + 1 \times 2^{-2} + 1 \times 2^{-3} = 1.375$.

The total number is therefore $-1 \times 1.375 \times 2^2 = -5.5$.

4. What is the 32-bit floating point representation of 17.75?

Answer:

(a) The sign is positive, hence $S = 0$.

(b) $17.75_{10} = 10001.11_2$. To normalize, we divide 10001.11 by 2^4 (to yield 1.000111). Therefore, the exponent is $E = 4 + 127 = 131_{10} = 10000011_2$.

(c) The mantisa is the resulting normalized form, starting from the first 1: $M = 1110\ 0000\ \ldots$.

All together this yields 17.75 =

0	1	0	0	0	0	0	1	1	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Problem 3: Data Representation

The English alphabet has 26 letters. Assuming they occur with equal probabilities:

1. What is the probability P_i of each letter?

Answer:
$P_i = \frac{1}{26}$

2. What is the entropy of the alphabet?

Answer:

$$
\begin{aligned}
S &= -\sum_{i=1}^{N} \left[p\left(x_i\right) \cdot \log_2 p\left(x_i\right)\right] \\
&= \sum_{i=1}^{N} \left[p\left(x_i\right) \cdot \log_2 p\left(\frac{1}{x_i}\right)\right] \\
&= \sum_{i=1}^{26} \left[\frac{1}{26} \cdot \log_2 26\right] \\
&= \frac{1}{26} \sum_{i=1}^{26} \log_2 26 \\
&= \cancel{\frac{1}{26} \cdot 26} \cdot \log_2 26 \\
&= \log_2 26 \\
&\approx 4.7
\end{aligned}
$$

3. If we encode the alphabet in 8 bits code, what is the redundancy?

Answer:
$R = H_0 - H = 8 - 4.7 = 3.3$

4. If the redundancy of the alphabet in ASCII is so big, why is 8-bit coding used for it?

Answer:
In ASCII the alphabet appears twice: once as uppercase letters (A-Z) and once as lowercase letters (a-z). In addition, ASCII encodes all the digits (0-9), several signs (i.e. /!#@) and control characters (i.e. start of text, cancel, acknowledgment, etc.). All in all, 256 characters are encoded, meaning $H = 8$ and a redundancy $R = 0$.

Problem 4: SQL Databases

An online forum for discussions about science and medicine has many users, each with his/hers own unique id, username, password, date of joining the forum, date of last message posted and preferred graphical theme. The data is stored in a relational database.

1. Write SQL code to generate the table 'users' which implements the structure above (keyed by user IDs).

Answer:

```sql
CREATE TABLE users (
  ID      INT(10) NOT NULL,
  username char(100),
  password char(256),
  join_date DATE,
  last_msg_date DATETIME,
  theme INT(10),
  PRIMARY KEY (ID)
)
```

2. Write a command to list the data for the first 100 users (i.e. $0 \leq \mathrm{ID} < 100$).

Answer:

```sql
SELECT * FROM users
  WHERE ID >= 0 AND ID < 100;
```

3. The 1,337th user to join was *DMendel*, password 'theMEndel'. He joined on January 30th 2003 and prefers the theme 'Tapirs'. Write a code to add him to the table.

Answer:

```sql
INSERT INTO users
  VALUES(1337,
         'DMendel',
         'theMEndel',
         '2003-01-30',
         NULL,
         'tapirs');
```

NOTE: It is really unadvised to store passwords in their 'plain text' format, since in case the database is leaked, user's accounts can be used directly. Instead, sensitive data like passwords is stored as hashed (encrypted) strings, using advanced hashing functions (like HSA256). Modern applications also use a method called 'Salting' to increase the security even further[a].

[a] See https://www.youtube.com/watch?v=8ZtInClXe1Q for more information.

4. *DMendel* wrote a new message. What command should be used to update this information? (assume NOW is a variable storing the current time)

Answer:

```sql
UPDATE users
  SET last_msg_date = NOW
  WHERE username = 'DMendel';
```

5. User *AnWakefi* decided that he did not like the forum and thus un-registered from it. How would you delete his account?

Answer:

```sql
DELETE FROM users
  WHERE username = 'AnWakeFi';
```

6. User *mariecur* wishes to see the date of her last post + which theme she is using. How would you implement this query?

Answer:

```sql
SELECT FROM users last_msg_date, theme
  WHERE username = 'mariecur';
```