HOMEWORK SET 6. CARDINALITY AND COUNTING

1. A particular brand of shirt comes in 8 colors, has a male version and a female version, and comes in four sizes for each sex. How many different types of this shirt are made?

Solution. We can think of a type of a shirt as an element of the Cartesian product of several sets: C (color), of cardinality 8, $V = \{F, M\}$, and S, of cardinality 4 (as a matter of fact, sizes can be different for male and female versions but for counting purposes it does not matter as long as there is the same number of sizes). Thus, there are $8 \cdot 2 \cdot 4 = 64$ different types. □

2. A model of car has a front-wheel drive (FWD) version and an all-wheel drive (AWD) version. Each version can be a hybrid, a gas or a diesel. An FWD version is always automatic while AWD can be either automatic or manual shift. Each car is of one of six different colors. How many different variants are there?

Solution. We can consider FWD and AWD versions as elements of two disjoint sets. Let $F = \{H, G, D\}$ be the set of fuel types (H stands for hybrid, G for gas and D for diesel). Let C be the set of possible colors; then $|C| = 6$. Finally, let $T = \{A, M\}$ be the set of transmission types. An FWD model is an element of the set $F \times C$ of cardinality $|F||C| = 3 \cdot 6 = 18$, and an AWD model is an element of the set $F \times C \times T$ of cardinality $|F||C||T| = 3 \cdot 6 \cdot 2 = 36$. Thus, the total number of different variants is $18 + 36 = 54$. □

3. A palindrome is a string whose reversal is identical to the string (for example, 0110 and 010010 are palindromes). How many bit strings of length n are palindromes?

Solution. If $n = 2k$ then for a bit string to be a palindrome, we must have the last k bits to be the first k bits written in reverse order. This means that the first k bits uniquely determine the remaining k bits and so there are as many palindromes of length $2k$ as there are bit strings of length k. Thus, there are 2^k palindromes. If $n = 2k + 1$ then the last k bits of our string must still be the first k written in the reverse order and the value of the $(k + 1)$th bit does not matter. So, in this case we have $2 \cdot 2^k = 2^{k+1}$ palindromes. Thus, the number of palindromes among bit strings of length n is $2^{\lceil n/2 \rceil}$ (here $\lceil x \rceil$ is the smallest integer k such that $x \le k$). □

4. How many bit strings of length 10 begin with two 0s but do not end with two 0s?

Solution. There are $2^{10-2} = 2^8 = 256$ bit strings of length 10 which begin with two 0s. Among those there are $2^{8-2} = 2^6 = 64$ which end with two 0s. Thus, there are $256 - 64 = 192$ bit strings satisfying both conditions (note that the total number of bit strings of length 10 is $2^{10} = 1024$). □

5. A computer system uses passwords which can only contain uppercase letters, lowercase letters and digits from 0 to 9. A password must be at least 6 and at most 8 symbols long.

a) Assuming that no further restrictions are imposed, how many different passwords can be made?

b) Suppose that a password must begin with a letter and must contain at least one digit. How many different passwords can be made?

c) Propose your own restrictions and find the number of different passwords in that case.

Solution. P_n be the number of passwords of length n. We can have $n = 6, 7, 8$ and the sets P_n are clearly disjoint. Let L be the set of lowercase letters, U be the set of uppercase letters and D be the set of digits from 0 to 9. Clearly $|L| = |U| = 26$ and $|D| = 10$.

a) Elements of P_n are taken from the nth Cartesian power of the set $L \cup U \cup D$ which has cardinality $26 + 26 + 10 = 62$. Thus, $|P_n| = 62^n$ and so $|P_6| = 56,800,235,584$, $|P_7| = 3,521,614,606,208$ and $|P_8| = 218,340,105,584,896$. The total number is $|P_6| + |P_7| + |P_8| = 221,918,520,426,688$.

b) Elements of P_n are taken from the Cartesian product of $L \cup U$ (the set of all letters, lower and upper case, of cardinality 52) and $(n-1)$th copies of $L \cup U \cup D$, of cardinality 62. However, we must exclude passwords which contain no digits, that is, which are elements of $(L \cup U)^n$. So, $|P_n| = 52 \cdot 62^{n-1} - 52^n = 52(62^{n-1} - 52^{n-1})$. This gives $|P_6| = 27,868,297,600$, $|P_7| = 1,925,540,547,840$ and $|P_8| = 129,664,230,991,360$ with the total of $131,617,639,836,800$.

c) A typical restriction is that a password must begin with a letter and must contain at least one lowercase letter, at least one uppercase letter and at least one digit. So, we need to remove from $(L \cup U) \times (L \cup U \cup D)^{n-1}$ all passwords which contain no uppercase letters (elements of $L \times (L \cup D)^{n-1}$) all passwords which contain no digits (elements of $(L \cup U)^{n-1}$) and all passwords which contain no lowercase letters (elements of $U \times (U \cup D)^{n-1}$). So

$$|P_n| = 52 \cdot 62^{n-1} - 26 \cdot 36^{n-1} - 52^n - 26 \cdot 36^{n-1} = 52(62^{n-1} - 36^{n-1} - 52^{n-1})$$

Thus, $|P_6| = 24,724,056,448$, $|P_7| = 1,812,347,866,368$, $|P_8| = 125,589,294,458,368$, with the total of $127,426,366,381,184$.

□

6. The International Telecommunications Union (ITU) specifies that a telephone number must consist of a country code with between 1 and 3 digits, except that the code 0 is not available for use as a country code, followed by a number with at most 15 digits. How many available possible telephone numbers are there that satisfy these restrictions?

Solution. The country code is an at most 3 digit number with 0 excluded, which gives 999 possible country codes.

A phone number is a 15-digit number, which gives 10^{15} possibilities. Thus, the total number is $999 \cdot 10^{15} = 9.99 \cdot 10^{17}$. □

7. Data are transmitted over the Internet in *datagrams*, which are structured blocks of bits. Each datagram contains header information organized into a maximum of 14 different fields (specifying, in particular, the source and destination addresses) and a data area that contains the actual data that are transmitted. One of the 14 header fields is the header length field (denoted by `IHL`, Internet Header Length) which is specified by the protocol to be 4 bits long and that contains the header length in 32-bit blocks of bits. For example, if `IHL` $= 0101$, the header is made up of five 32-bit blocks. Another of the 14 header fields is the 16-bit-long total length field (denoted by `TL`), which specifies the length in octets (groups of 8 bits) of the entire datagram, including both the header fields and the data area. The length of the data area is the total length of the datagram minus the length of the header.

a) The largest possible value of `TL` (which is 16 bits long) determines the maximum total length in octets of an Internet datagram. What is this value?

b) The largest possible value of `IHL` (which is 4 bits long) determines the maximum total header length in 32-bit blocks. What is the maximum total header length in octets?

c) The minimum (and most common) header length is 20 octets. What is the maximum total length in octets of the data area of an Internet datagram?

d) How many different strings of octets in the data area can be transmitted if the header length is 20 octets and the total length is as long as possible?

Solution. **a)** `TL` stores the length of the datagram in octets. Since it is 16 bits long, the maximal value of `TL` is $2^{16} - 1 = 65,535$ octets. This is the maximal possible length of a datagram (in octets or bytes).

b) The maximal header length in 32-bit blocks is $2^4 - 1 = 15$. Since each 32-bit block consists of $32/8 = 4$ octets, the maximal possible length of the header is $15 \cdot 4 = 60$ octets.

c) Since the minimal length of the header is 20 octets and the maximal total length of a datagram is $65,535$, the maximal length of the data area is $65,535 - 20 = 65,515$ octets.

d) An octet can have $2^8 = 256$ possible values. So, there can be at most $256^{65,515} = 2^{8(2^{16}-21)}$ different data strings. This number is approximately $7 \cdot 10^{157,775}$.

□

Remark. The full datagram header structure in IPv4 is shown on the diagram below

<table>
<tr><td>Bit number</td><td colspan="5">0 4 8 12 16 19 20 24 28 32</td></tr>
<tr><td>Block 1</td><td>Version</td><td>IHL</td><td>Type of service (TOS)</td><td colspan="2">Total length (TL)</td></tr>
<tr><td>Block 2</td><td colspan="3">Identification</td><td>Flags</td><td>Fragment offset</td></tr>
<tr><td>Block 3</td><td colspan="2">Time to live (TTL)</td><td>Protocol</td><td colspan="2">Header checksum</td></tr>
<tr><td>Block 4</td><td colspan="5">Source address</td></tr>
<tr><td>Block 5</td><td colspan="5">Destination address</td></tr>
<tr><td>Block 6</td><td colspan="5">Options</td></tr>
<tr><td>⋮</td><td colspan="5">⋮</td></tr>
<tr><td>Block n</td><td colspan="4"></td><td>Padding</td></tr>
</table>

Here $n \leq 15$ and blocks with numbers ≥ 6 are optional. Version has value 4 for IPv4. `TOS` is seldom used. Flags are used to indicate whether the message was fragmented. Identification is used to identify fragments of the same message, that is, this field has the same value for all fragments of the same message (IPv4 is *asynchronous*, that is, a message can arrive in fragments in mixed order and even from different sources). Time to live specifies the number of routers that a message can possibly go through (each router subtracts one from this field; when its value is zero the message is discarded). Protocol identifies the low-level protocol used in this message (such as TCP/IP, UDP, ICMP etc). Header checksum is calculated by splitting the header in 16-bit words and adding them together. As an exercise, find the maximal value of `TTL` and the maximal number of fragments a message can have (for the second one you need to know that the *position* of a fragment in a message is specified in a 13-bit long field "Fragment offset" in terms of octets).

8. In the 17th century, there were more than $800,000$ inhabitants of Paris. At the time, it was believed that no one had more than $200,000$ hairs on their head. Assuming these numbers are correct and that everyone has at least one hair on their head (that is, no one is completely bald), show that there had to be at least two Parisians with the same number of hairs on their heads.

Solution. We can use the pigeonhole principle. The boxes are labeled by the number of hairs a person has on his or her head. Then we have $200,000$ boxes and $800,000$ "pigeons" to put in these boxes. □

9. Prove that, given three integers, there are always at least two among them whose sum is even, without analyzing all possible cases.

Solution. By the pigeonhole principle, two integers out of any collection of 3 integers have the same parity. It remains to observe that the sum of two integers of the same parity is even. □

10. Use induction to prove that if sets $S_1, \ldots, S_n$ are pairwise disjoint (that is, $S_i \cap S_j = \emptyset$ if $i \neq j$) then $|S_1 \cup \cdots \cup S_n| = |S_1| + \cdots + |S_n|$.

Solution. The induction base here is the case when $n = 2$. Then $|S_1 \cup S_2| = |S_1| + |S_2|$ by the first basic counting principle.

For the inductive step, we claim that if our sets are pairwise disjoint then $(S_1 \cup \cdots \cup S_n) \cap S_{n+1} = \emptyset$. Then by definition we have $|S_1 \cup \cdots \cup S_n \cup S_{n+1}| = |S_1 \cup \cdots \cup S_n| + |S_{n+1}| = (|S_1| + \cdots + |S_n|) + |S_{n+1}| = |S_1| + \cdots + |S_{n+1}|$.

Thus, it remains to prove that $(S_1 \cup \cdots \cup S_n) \cap S_{n+1} = \emptyset$. Indeed, assume that $x \in (S_1 \cup \cdots \cup S_n) \cap S_{n+1}$. Then $x \in S_1 \cup \cdots \cup S_n$ and $x \in S_{n+1}$. The first statement means that $x \in S_i$ for some $1 \leq i \leq n$. Thus, $x \in S_i \cap S_{n+1} = \emptyset$, which is a contradiction.

$\square$