**MOCK-UP Exam: LOGIC / Problem Solving**

**BATCH 2024**

**{2h 00}**

25 March 2023

FIRST NAME:

LAST NAME:

CLASS:

* The allowed documents are
  + 1 single A4 sheet of paper with notes
  + The binary code flippy chart
* Chatting and talking to other students are forbidden

|  |  |
| --- | --- |
| EXERCICES | POINT |
| Exercise 1 | 10 |
| Exercise 2 | 15 |
| Exercise 3 | 25 |
| Exercise 4 | 25 |
| Exercise 5 | 25 |
| **TOTAL** | **100** |

**Exercise 1:(10pts)**

The alphabet is given below to help you:

a b c d e f g h i j k l m n o p q r s t u v w x y z

**Q1** What is the missing letter in this series? នៅក្នុងជួរខាងក្រោមនេះ​ តើអ្នកបាត់អក្សរអ្វី?​ (2pts)

a z b ? c x

Your answer: **y**

**Q2** What is the missing letter in this series? (4pts)

a c f ? o u

Your answer: **j**

**Q3** What is the missing letter in this series? (4pts)

DEF DEF2 DE2F2  ? D2E2F3

Your answer: **D2E2F2**

**Exercise 2:(15pts)**

**Q1** How many shades of color are possible with 3 bits? Write down the possible colors(4pts)

There are **8 colors**. **000,001,010,011,100,101,110,111**

**Q2** How do I calculate the size in megabytes of an RGB image of size 1024 x 1024? (5pts)

Bits of rgb color = 8+8+8 = 24bits.

=>**size in bits** = 1024x1024x24 bits.

But 1byte = 8bits=>bits = 1/8byte

=>**size in byte** = (1024x1024x24)/8 byte

But 1kilobyte = 1024byte =>byte = 1kilobyte/1024

=>**size in kilobyte** = (1024x1024x24)/(1024x8) kilobyte

But 1megabyte = 1024Kilobyte =>kilobyte = 1/1024 megabyte

=>**size in megabyte** = (1024x1024x24)/(1024x1024x8) megabyte

= **3megabytes.**

**Q3** What is the **output color** rgb(0,0,255)?(2pts)

Output **Blue** color.

**Q4** What is the **Hex color** of rgb(0,192,255)? (4pts)

0 = 00 and 192 = C0 and 255 = FF

**So rgb(0,192,255) = #00C0FF**

**Exercise 3:(25pts)**

**Q1** What is the result of this operation with binary numbers? ​ តើចម្លើយរបស់អ្នកស្មើប៉ុន្មាន បន្ទាប់ពីការធ្វើប្រមាណវិធីដក​លេខប្រព័ន្ធគោលពីរ?(7pts)

111101

- 010011

- 011010

Your answer: **10000**

**Q2** What is the result of this operation with hexadecimal numbers? ​ តើចម្លើយរបស់អ្នកស្មើប៉ុន្មាន បន្ទាប់ពីការធ្វើប្រមាណវិធីបូក​លេខប្រព័ន្ធគោលដប់ប្រាំមួយ?(9ptspts)

A4F8

+ FF44

+ BCDE

Your answer: **2611A**

**Q3** What is the result of this operation with hexadecimal numbers? តើចម្លើយរបស់អ្នកស្មើប៉ុន្មាន បន្ទាប់ពីការធ្វើប្រមាណវិធីដក​លេខប្រព័ន្ធគោលដប់ប្រាំមួយ? (9pts)

FABA

- B0A

- 7BD

Your answer: **E7F3**

**Exercise 4:(25pts)**

Compute the following conversions

|  |  |
| --- | --- |
| Base 16 | Base 10 |
| EF5 (5pts)  10F (5pts) | 3829  271 |

|  |  |
| --- | --- |
| Base 10 | Base 8 |
| 78 (3pts) 99 (3pts) | 116  143 |

|  |  |
| --- | --- |
| Base 2 | Base 16 |
| 10000011110000 (4pts)  11111100001101010 (5pts) | 20F0  1F86A |

**Exercise 5:(25pts)**

*On my computer, I want to store pictures – Each picture size is 3* ***bytes****.*

**Q1.** What is the size **in bits** of one picture? Explain why (5pts)

1byte = 8bits

=>3bytes = 8x3bits = 24bits

So, **size in bits is 24bits of one picture**

**Q2.** I want to store 12 pictures. What computer memory size do I need? (In bits) Explain why. (3pts)

1picture = 24bits

12pitures = bits?

=>bits = (12picture x 24 pictures)/1picture = **288 bits.**

*On my computer, I want store numbers – Each number is defined from 0 to 12*

**Q3.** What is the size **in bits** of one number? Explain why. (5pts)

*number is defined from 0 to 12*, 23 = 8 not enough and 24 = 16

=>4 bits in one number.

So, **we have 4bits in one number**.

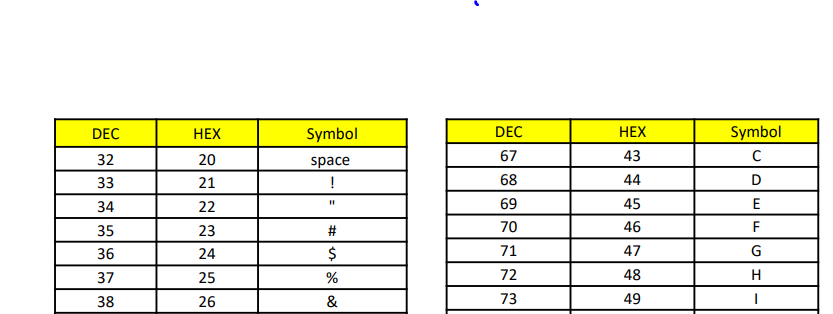
**Q4.** My computer memory size is 52 bits. How many numbers can I store? Explain why. (3pts) we have 1number = 4bits

Number? = 52bits

=>number = (1number x 52bits)/4bits = 13numbers

So, **We need 13numbers with 52bits**

**Q5.** We have ASCII printable characters as below.



* What is the **binary** representation for the word **HI ! (5pts)**

H = 4816 = 010010002 and I = 4916 = 010010012 and space = 2016 = 001000002 and ! = 2116 = 001000012

**So HI ! = 01001000 01001001 00100000 00100001**

* Convert this binary to hexadecimal and then to ASCII. (4pts)

00100001 00100100 00100000 00100100 00100001

00100001 **= 2116 = !** and 00100100 **= 2416 = $** and 00100000 **= 2016 = space** and 00100100 **= 2416 = $** and 00100001 **= 2116 = !**

So **00100001 00100100 00100000 00100100 00100001 = !$ $!**