**Subject: PRF192- PFC**

**Workshop 01**

**Objectives:**

1. Reviewing for number systems
2. Exploring memory of a C program

**Recommendations**

Part 1: Students do exercises using notebooks

Part 2: Students develop programs, run them, write down their memory structure to notebooks.

**Part 1: Number systems**

**Exercise 1** **(2 marks): Convert decimal numbers to binary ones**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Decimal** | **4-bit Binary** | **Decimal** | **8-bit Binary** | **Decimal** | **16-bit Binary** |
| 9 | 1001 | 7 | 00000111 | 255 | 0000 0000 1111 1111 |
| 7 | 0111 | 34 | 00100010 | 192 | 0000000011000000 |
| 2 | 0010 | 125 | 01111101 | 188 | 0000000010111100 |
| 15 | 1111 | 157 | 10011101 | 312 | 0000000100111000 |
| 12 | 1100 | 162 | 10100010 | 517 | 0000001000000101 |
| 11 | 1011 | 37 | 00100101 | 264 | 0000000100001000 |
| 6 | 0110 | 66 | 01000010 | 543 | 0000001000011111 |
| 5 | 0101 | 77 | 01001101 | 819 | 0000001100110011 |
| 8 | 1000 | 88 | 01011000 | 1027 | 0000010000000011 |
| 13 | 1101 | 99 | 01100011 | 2055 | 0000100000000111 |
| 14 | 1110 | 109 | 01101101 | 63 | 0000000000111111 |

**Exercise 2(2 marks): Convert decimal numbers to binary and hexadecimal ones**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Decimal** | **Binary** | **Hexa.** | **Decimal** | **16-bit Binary** | **Hexadecimal** |
| 9 | 1001 | 9 | 255 | 0000 0000 1111 1111 | 00FF |
| 127 | 0111 1111 | 9F | 192 | 0000 0000 1100 0000 | 00C0 |
| 125 | 0111 1101 | 7D | 188 | 0000 0000 1011 1100 | 00BC |
| 157 | 1001 1101 | 9D | 312 | 0000 0001 0011 1000 | 0138 |
| 162 | 1010 0010 | A2 | 517 | 0000 0010 0000 0101 | 0205 |
| 37 | 0010 0101 | 25 | 264 | 0000 0001 0000 1000 | 0108 |
| 66 | 0100 0010 | 42 | 543 | 0000 0010 0001 1111 | 021F |
| 77 | 0100 1101 | 4D | 819 | 0000 0011 0011 0011 | 0333 |
| 88 | 0101 1000 | 58 | 1027 | 0000 0100 0000 0011 | 0403 |
| 99 | 0110 0011 | 63 | 2055 | 0000 1000 0000 0111 | 0807 |
| 109 | 0110 1101 | 6D | 63 | 0000 0000 0011 1111 | 003F |

**Exercise 3(2 marks): Compute**

(b: binary, q: octal, h: hexadecimal)

**3245q + 247q = ?3514q = ?0111 0100 1100b**

**1A7Bh + 26FE7h = ?28A62h = ?0010 1000 1010 0110 0010b**

**1101101101b - 10110111b =?0010 1011 0110 b**

**3654q – 337q =?3315q = ?0110 1100 1101b**

**3AB7h – 1FAh = ?38BDh = ?0011 0100 1011 1101 b**

**36Ah – 576q = ?1 ECh = ?0001 1110 1100 b**

**64AEh – 1001101b= ? 62141q**

101101111 b

+ 100111011 b

110110001 b

110001101 b

**0101 1110 1000** b

1011010 b\* 1011b = **0011 1101 1110** b

1101000b + 2AB h + 345 q = ?**3F8** h = ?**1770** q

3AFh / 1Ch =?**0010 0001** b = ?**33**d

3ACh – 562q = ?**0010 0011 1010** b = ?**570** d

3FFA h / 327q = ?**0100 1100**b = ?**76** d

**Exercise 4 (2 marks)**

1. Show binary formats of 1-byte unsigned numbers: 251 , 163, 117

251: 1111 1011  
163: 1010 0011  
117: 0111 0101

1. Show binary formats of 2-byte unsigned numbers: 551 , 160, 443

551: 0000 0010 0010 0111  
160: 0000 0000 1010 0000  
443: 0000 0001 1011 1011

1. Show binary formats of 1-byte signed numbers: -51 , -163, -117, 320

-51: 1011 0011  
-163: ~~1000 000~~ 1010 0011  
-117: 1111 0101  
320: ~~0001~~ 0100 0000

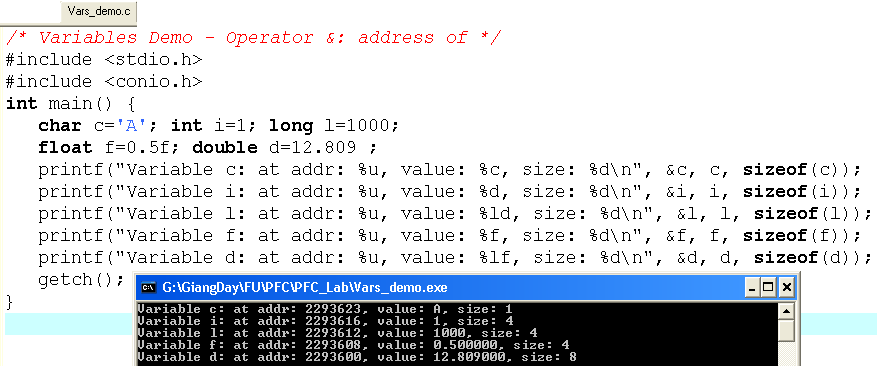
1. Show the decimal values of 1-byte unsigned representations: :

01100011 b , 10001111 b , 11001010 b , 01001100 b

01100011: 99  
10001111: 143  
11001010: 202  
01001100: 76

**Part 2: Explore memory structure of programs**

**Sample**



12.809

0.5

1000

‘A’

1

**d:2293600**

**f:2293608**

**l:2293612**

**i:2293616**

**c:2293623**

**Complete the code of following program then draw it’s memory structure**

**(2 marks)**

