Lab 1

# Part 1

1)

|  |  |  |
| --- | --- | --- |
| a)10 | c)64 | e)FF |
| b)40 | d)65 | f)E01 |

2)

|  |  |  |
| --- | --- | --- |
| a)1010 | c)1100111 | e)11111111 |
| b)1000001 | d)110001000 | f)10000000001 |

3)

|  |  |  |
| --- | --- | --- |
| a)8 | c)17 | e)240 |
| b)11 | d)85 | f)513 |

4)

|  |  |  |
| --- | --- | --- |
| a)A | c)99 | e)8D5 |
| b)D | d)C7 | f)BADE |

5)

|  |  |  |
| --- | --- | --- |
| a)1110 | c)101000000 | e)C1 |
| b)100000001 | d)10A | f)10A1 |

# Part 2

3)

|  |
| --- |
| Python 3.9.6 (v3.9.6:db3ff76da1, Jun 28 2021, 11:14:58)  [Clang 12.0.5 (clang-1205.0.22.9)] on darwin  Type "help", "copyright", "credits" or "license()" for more information.  >>> value = 1  >>> value + 5  6  >>> print(value)  1  >>> value=value+7  >>> value = value - 4  >>> value  4  >>> message = "Hello, "  >>> print(message, "World!")  Hello, World!  >>> print(value + 10, abs(value - 10))  14 6  >>> print(Value)  Traceback (most recent call last):  File "<pyshell#9>", line 1, in <module>  print(Value)  NameError: name 'Value' is not defined  >>> 5 = value  SyntaxError: cannot assign to literal  >>> (value+1)//2  2 |

4)

|  |
| --- |
| Python 3.9.6 (v3.9.6:db3ff76da1, Jun 28 2021, 11:14:58)  [Clang 12.0.5 (clang-1205.0.22.9)] on darwin  Type "help", "copyright", "credits" or "license()" for more information.  >>> value = 7  >>> value + 2  9  >>> value - 2  5  >>> -value  -7  >>> value \* 2  14  >>> value / 2  3.5  >>> value // 2  3  >>> value \*\* 2  49  >>> value % 4  3  >>> value + 2 \* value - 2  19  >>> (value + 2) \* (value - 2)  45  >>> |

5a)

|  |
| --- |
| >>> result = input("What now? ")  What now? bruh  >>> print (result)  bruh |

5b)

|  |
| --- |
| >>> x = -5  >>> y=2  >>> print(min(x, y))  -5  >>> print(max(x, y))  2  I think min finds the smallest number while max finds the biggest one |

5c)

|  |
| --- |
| >>> x = 1/3  >>> y = 2/3  >>> print(round(x), round(y))  0 1  >>> print(round(x, 2), round(y, 2))  0.33 0.67  >>> print(x, y)  0.3333333333333333 0.6666666666666666  It doesn’t change the values but it specifies the number of decimals that it is rounded to |

6a)

|  |
| --- |
| >>> print(2031408 % 2)  0 |

6b)

|  |
| --- |
| >>> print(2031408 % 100)  8 |

6c)

|  |
| --- |
| >>> print(2031408 % -100)  -92 |

6d)

|  |
| --- |
| They are related as the divisor are the opposite signs of each other. The answers however relate to each other as they are both being divided by 100 but in different sign contexts. |

7

|  |
| --- |
| >>> x = 2031408  >>> y = 1000  >>> abs(y \* y - x) < 1e-3  False  >>>  >>> y = (y + x / y) / 2  >>> abs(y \* y - x) < 1e-3  False  >>> print("The square root of", x, "is", y)  //Prints in shell if its within the parameters for it to be a true answer  The square root of 2031408 is 1515.704  >>> y = (y + x / y) / 2  >>> abs(y \* y - x) < 1e-3  False  >>> y = (y + x / y) / 2  >>> abs(y \* y - x) < 1e-3  False  >>> y = (y + x / y) / 2  >>> abs(y \* y - x) < 1e-3  True  >>> print("The square root of", x, "is", y)  The square root of 2031408 is 1425.2747103651604  >>> |

8)

|  |
| --- |
| Python 3.9.6 (v3.9.6:db3ff76da1, Jun 28 2021, 11:14:58)  [Clang 12.0.5 (clang-1205.0.22.9)] on darwin  Type "help", "copyright", "credits" or "license()" for more information.  >>>  = RESTART: /Users/gordonng/Library/Mobile Documents/com~apple~CloudDocs/Documents/Cegep/Introduction to Programming/id\_sqrt.py  The square root of 2031408 is 1425.2747103651604  >>>  It differs as it only gives the end result |

Copy of id\_sqrt.py

x = 2031408

y = 1000

y = (y + x / y) / 2

abs(y \* y - x) < 1e-3

y = (y + x / y) / 2

abs(y \* y - x) < 1e-3

y = (y + x / y) / 2

abs(y \* y - x) < 1e-3

y = (y + x / y) / 2

abs(y \* y - x) < 1e-3

print("The square root of", x, "is", y)