

CS2910 Exercise: UDP Encoding

Manipulating Decimal Digits

Create a python file called "digits.py." In that file, write the following methods:

1. `find_lowest_two_digits(input)`
Given a decimal number, return a decimal number with the lowest two digits of the given number. For example, given 125834, return 34.
2. `create_1nn_number(input)`
Given a decimal number, return 1XX, where XX are the third and fourth lowest digits of the number. For example, given 125834, return 158. You can use `find_lowest_two_digits(input)` if you like.
3. `create_ascii_string(input)`
Given a decimal number, compute an ASCII string for that number. You can convert an ASCII character to a number using `ord('A')`, and you can convert the number back to an ASCII character using `chr(ascii_code)`
4. `main()`
Write a main function to demonstrate operation of your other methods. Invoke it at the end of your program.

In a comment in your code, answer this question:

How would you change your code if you wanted to manipulate binary numbers instead of decimal? How about hexadecimal?

Manipulating Text Data

Create a file called "text.py." In that file, write the following methods:

1. `convert_string_to_hex(input_string)`
Accept a normal Python string as argument, and convert it to a string of hex characters representing the bytes in the string. Return the resulting hex string from the function.
For example, `convert_string_to_hex('Ab3')` should return `'416233'`.
2. `find_integer_length_for_value(input_value)`
Accept a non-negative integer value and return the number of bits needed to represent it as an unsigned binary value.
For example, `find_integer_length_for_value(9)` should return 4.
3. `make_utf8_string(code_point)`
Accept a non-negative integer representing a Unicode code point and return a string containing the bytes needed to encode it in UTF-8. You may not use a Python built-in or library function to do the conversion.
For example, `make_utf8_string(0x2615)` should return the equivalent of `chr(0xE2)+chr(0x98)+chr(0x95)`. Use `convert_string_to_hex(input_string)` to confirm your method works.
4. `main()`
Write a main function to demonstrate operation of your other methods. Invoke it at the end of your program.

Turn-in instructions will be announced in class on Monday. This exercise will not be graded.