請照題號次序作答

Please use C, C++, Java or Python programming language to design your computer programs.

1. (20%) Given a string consist of "(" and ")" symbols, for every right parenthesis there exists a matching left parenthesis, we said that string is a **balanced parentheses sequence.** For example, "()" , "(()" , "((()))" , ")(((()))" are balanced parentheses sequence. On the other hand, ")(" , "(()))(" and "()(()(" are not balanced parentheses sequence. Please implement a function `balanced_parenthese(seq)` which takes one string `seq` as the parameter. The function will return **true** if the string `seq` is a balanced parentheses sequence or **false** otherwise.

2. (20%) Huffman coding
   a. (10%) According to the sentence **"this is an example of a huffman tree"**, please generate the Huffman code table and encode the above sentence with your code table.
   b. (10%) Given a string S which consist of space char and lower case letters (i.e.,' ', 'a', ..., 'z'), implement a function `huffman_code(str)` which can output the huffman code table which is optimized to the input string S and convert the `str` into a bit pattern according to the generated huffman code table.

3. (20%) A **sum-product number** in a given number base b is a natural number that is equal to the product of the sum of its digits and the product of its digits. For example, the number 144 in base 10 is a sum-product number, because 1 + 4 + 4 = 9, (1)(4)(4) = 16, and (9) (16) = 144. Please implement a function `is_sum_product_number(x)` to check whether a natural number x is a **sum-product number** in base 10.

4. (20%)
   a. (10%) Please implement the **heapsort** algorithm. `heapsort(list, n)`. `list`: an array or a list, `count`: the number of data in the `list` that needs to be sort.
   b. (5%) Is **heapsort** a stable sorting algorithm?
   c. (5%) What is the time complexity of heapsort in **worst case**?

5. (20%) Explain the meanings of the following terminologies.
   a. **Coupling** and **Cohesion**
   b. **Abstract Data Type** and **Encapsulation**
   c. **Machine Cycle** and **Pipeline**
   d. **Virtual Memory** and **Virtual Address**

試題隨卷繳回