COVENANT UNIVERSITY
CANAANLAND, KM 10, IDIROKO ROAD
P.M.B 1023, OTA, OGUN STATE, NIGERIA.
B. Eng. DEGREE EXAMINATION

COLLEGE: ENGINEERING **DEPARTMENT:** PETROLEUM ENGINEERING
SESSION: 2022/2023 **SEMESTER:** OMEGA
COURSE CODE: PET328 **CREDIT UNIT:** 2
COURSE TITLE: COMPUTER APPLICATIONS IN PETROLEUM ENGINEERING **TIME:** 2 HOURS
INSTRUCTIONS: Answer ALL questions.

Congratulations on your recent placement as a Reservoir Engineering Intern! The Senior Reservoir Advisor (SRA) and the Lead Software Developer (LSD) are your training mentors, and have tasked you with the following requests. A custom Python module, *peteng.py* being built and used by the TTOWG Asset Team is herewith attached.

Question 1: [17½ marks]

a. Given that you are developing a script to be used by members of the team, write a Python statement to request an ID string from a user. Why would it be unnecessary to convert the user's response to a string? **[2 marks]**

b. Write a Python statement to make Module *peteng* available in your script, in anticipation of needing most of the functions therein. Take note that it is desirable to avoid prefixing with file name. **[2 marks]**

c. Extend the capability of Module *peteng* by including Function *archie_sw* according to the equation $s_w = \sqrt[n]{\frac{aR_w}{\phi^m R_t}}$. You may pass all parameters as function arguments. Let the return value be in 4 decimal places. **[3½ marks]**

d. Respond to the following questions from the LSD who is testing your understanding of Module *peteng*.
   i. What is the implication, for users, of defaulting some arguments of *gas_density* (Line 8)? **[2 marks]**
   ii. What happens if a user calls Function *fvf* (Line 41) without specifying value for Argument *pb*? **[2 marks]**
   iii. Create an object to be passed as argument to Function *stoiip_2* (Line 72), use arbitrary values. **[2 marks]**
   iv. Why is Function *str* (Line 118) invoked?
   v. What is the essence of Lines 133 and 134? **[2 marks]**

Question 2: [17½ marks]

a. Give a sequence of GitHub operations you would need to implement in order to have offline access to project files in a repository that the SRA just shared with you. **[2 marks]**

b. Write a Python code to check that a user's input is numeric, and to display "Numeric input required" in case the user's input is non-numeric. **[2 marks]**

c. Create Function *hydrostat_p* that would compute drilling mud hydrostatic pressure at a given true vertical depth, D, according to the following algorithm:
$HP = 0.52\rho_m D$ if mud densiity, ρ_m is known, or, $HP = \delta_m D$ if pressure gradient, δ_m, is known. **[3½ marks]**

d. The SRA requests for a new reservoir volumetric parameter, gross reservoir volume (GRV), based on trapezoidal rule. Using the following algorithm, create Function *grv* to accept owc, top and $area\_list$ as arguments and returns vol. The terms a_o and a_n refers to the first and the last elements of *area_list* while other elements are denoted as a_1, a_2, ... a_{n-1}. Be aware that n is the number of elements in *area_list*. **[10 marks]**

$$GRV = h\left(\frac{1}{2}[a_o + a_n] + [a_1 + a_2 + \cdots + a_o + a_{n-1}]\right)$$

$$h = \frac{owc - top}{n}$$

Question 3: [17½ marks]

a. Given that 0.26 and 850 have been assigned to '*poro*' and '*perm*', respectively. Create a list of tuples, with each tuple holding a pair of the variable name and value. Convert the list to a dictionary object. **[2 marks]**.

b. Re-construct the following statement in a way that the conditions are based on the "*less than or equal to*" Boolean operator: `if co2_comp > 0.12 or n2_comp > 0.03 or h2s_comp > 0.` **[2 marks]**

c. Well *CH_23* is to be drilled in successive depth segments; cost of drilling a segment is: $cost = B + C_r(t + T)$ Construct a loop to compute the cost per segment and to obtain (by summation) the aggregate/cumulative cost, until the total drilled depth exceeds 10,000 ft. You may denote the depth per segment as *delta_D*. **[3½ marks]**

d. Write a script to implement the following workflow, in response to a request from the Reservoir Advisor:

- Make Function *stoiip_discretized* from Module peteng (Line 77) available for call. **[2 marks]**
- Create a Function *trimmed_sum* that receives *inp_list* and *threshold_val* (a float) and returns the sum of all elements of *inp_list* whose values are greater than *threshold_val* **[4 marks]**
- Call Function *stoiip_discretized* and pass the *stoiip_list* component of its output as argument to a call of Function *trimmed_sum.* **[4 marks]**

Question 4: [17½ marks]

a. What do you think would go wrong if Line L4a below is to be replaced with Line L4b, as per Function *gas-density* (Line 8 of *peteng*)?

```
gas_density(gravity = 0.786, pressure = 14.7, temperature = 520, z = 1) - - - - L4a
gas_density(0.786, 520) - - -- - - L4b
```

b. If you are to replace the list in Lines 87 and 96 of *peteng* with a tuple object, how would you implement the *.append* effect, considering that tuples are immutable and do not have the *.append* method? **[2 marks]**

c. Create Function *darcy_rate* according to the equation $q = C_f \frac{KA\Delta P}{\mu L}$. You may pass all parameters as function arguments. Parameter C_f is the conversion factor and is often (but sometimes not) equal to 0.001127. Let the return value be in 2 decimal places. **[3½ marks]**

d. For a given discretized reservoir model, the permeability values of some gridblocks are so low that the SRA would like gridblocks with permeability lower than a given cut-off value to be classified as 'inactive' while others are to be classified as 'active'. Create Function *block_classifier* that receives *perm_list* and *cut_off* (a float) and returns a dictionary of *block_ID* (as keys) and string '*inactive*' or '*active*' (as values). Keys *block_ID* should contain string 'Block' and the natural ordering of the blocks (e.g. 'Block1') **[10 marks]**

```
print("Best Wishes!!!")
```