



Answer All Questions (use any programming language.)

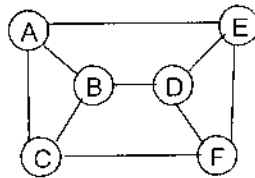
Question One

1- The following table show the hourly cost of four types of manufacturing processes. It also shows the number of hours required of each process to produce three different products. Use flow chart or Pseduo code (or any programming lag.) to solve the following:

- Determine the cost to make one unit of each product
- Suppose we produce 10 units of product 1, 5 of product 2 and 7 of product 3, Compute the total cost.

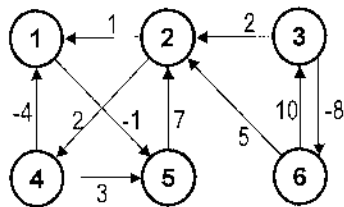
Process	Hourly Costs (\$)	Required hours to produce one unit Product 1	Required hours to produce one unit Product 2	Required hours to produce one unit Product 3
Lathe	10	6	5	4
Grinding	12	2	3	1
Milling	14	3	2	5
Welding	9	4	0	3

2- Draw the Breadth-First Search suitable data structure that result from a Breadth-First Search of the following graph starting with node A as the source.



Question Two

- 1- Give the differences between Big O, Theta and Omega complexity?
- 2- According to the following graph :Write adjacency matrix for this graph and **Adjacency List** . Then find the complexity at each case.



3- A supervisor in the same manufacturing company wants to produce a report from the PRODUCTION input file showing bonuses she is planning to give based on this year's production. She wants to have a report with three columns: last name, first name, and bonus. The bonuses is distributed as follows.

If this year's production is:

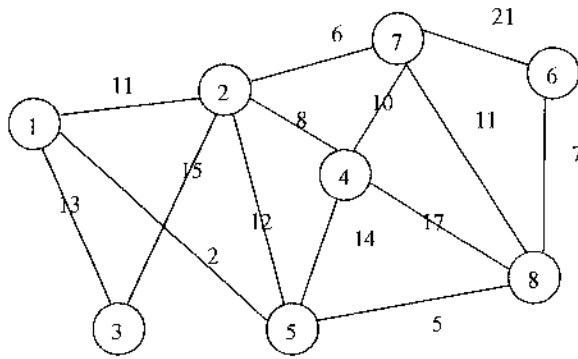
- 1,000 units or fewer, the bonus is \$25
- 1,001 to 3,000 units, the bonus is \$50
- 3,001 to 6,000 units, the bonus is \$100
- 6,001 units and up, the bonus is \$200

Create the flowchart or pseudocode for this program.

Question Three

1- Our coins come in units of 1 cent, 5 cents, 10 cents, 20 cents, 50 cents and 1 dollar. Assuming that there are limited numbers of coin as follows 100, 500, 500, 200, 200 and 1000 respectively , how would you derive a **coin-change algorithm** to compute the **minimum** number of coins required to make up a particular amount? For example, for 46 cents you need 4 coins: two 20-cent coins, one 5-cent coin, and one 1-cent coin.(Write pseudo code or flow chart for the proposed algorithm)

2- Compute a minimum cost spanning tree for the following graph using Prim's Algorithm or Kruskal's Algorithm



Find the time complexity of the above algorithm

2- Develop an algorithm to implement a system for generating 16 number for mobile charging cards . Consider the constrains (Number reuse, Fabricated number , collision with other numbers

Question Four (MCQ)

1. Is Python case sensitive when dealing with identifiers?
 - a) yes
 - b) no
 - c) machine dependent
 - d) none of the mentioned
2. What is the maximum possible length of an identifier?
 - a) 31 characters
 - b) 63 characters
 - c) 79 characters
 - d) none of the mentioned
3. Which of the following is invalid?
 - a) `_a = 1`
 - b) `__a = 1`
 - c) `__str__ = 1`
 - d) none of the mentioned
4. Which of the following is an invalid variable?
 - a) `my_string_1`
 - b) `1st_string`
 - c) `foo`
 - d) `_`
5. Why are local variable names beginning with an underscore discouraged?
 - a) they are used to indicate a private variables of a class
 - b) they confuse the interpreter
 - c) they are used to indicate global variables
 - d) they slow down execution
6. Which of the following is not a keyword?
 - a) `eval`
 - b) `assert`
 - c) `nonlocal`
 - d) `pass`
7. All keywords in Python are in _____
 - a) lower case
 - b) UPPER CASE
 - c) Capitalized
 - d) None of the mentioned
8. Which of the following is true for variable names in Python?
 - a) unlimited length
 - b) all private members must have leading and trailing underscores
 - c) underscore and ampersand are the only two special characters allowed
 - d) none of the mentioned
9. Which of the following is an invalid statement?
 - a) `abc = 1,000,000`
 - b) `a b c = 1000 2000 3000`
 - c) `a,b,c = 1000, 2000, 3000`
 - d) `a_b_c = 1,000,000`

10. Which of the following cannot be a variable?
- `__init__`
 - `in`
 - `it`
 - `on`
11. What is the output of `print 0.1 + 0.2 == 0.3`?
- True
 - False
 - Machine dependent
 - Error
12. Which of the following is not a complex number?
- `k = 2 + 3j`
 - `k = complex(2, 3)`
 - `k = 2 + 3l`
 - `k = 2 + 3J`
13. What is the type of `inf`?
- Boolean
 - Integer
 - Float
 - Complex
14. What does `-4` evaluate to?
- 5
 - 4
 - 3
 - +3
15. What does `~~~~~5` evaluate to?
- +5
 - 11
 - +11
 - 5
16. Which of the following is incorrect?
- `x = 0b101`
 - `x = 0x415`
 - `x = 19023`
 - `x = 03964`
17. What is the result of `cmp(3, 1)`?
- 1
 - 0
 - True
 - False
18. Which of the following is incorrect?
- `float('inf')`
 - `float('nan')`
 - `float('56'+78')`
 - `float('12+34')`
19. What is the result of `round(0.5) - round(-0.5)`?
- 1.0
 - 2.0
 - 0.0
 - None of the mentioned
20. What does `3 ^ 4` evaluate to?
- 81
 - 12
 - 0.75
 - 7
21. Which of the following statements create a dictionary?
- `d = {}`
 - `d = {"john":40, "peter":45}`
 - `d = {40:"john", 45:"peter"}`
 - All of the mentioned

22. What will be the output of the following Python code snippet? `d = {"john":40, "peter":45}`
- `"john", 40, 45, and "peter"`
 - `"john"` and `"peter"`
 - 40 and 45
 - `d = (40:"john", 45:"peter")`
23. What will be the output of the following Python code snippet?
`d = {"john":40, "peter":45} "john" in d`
- True
 - False
 - None
 - Error
24. What will be the output of the following Python code snippet?
`d1 = {"john":40, "peter":45} d2 = {"john":466, "peter":45} d1 == d2`
- True
 - False
 - None
 - Error
25. What will be the output of the following Python code snippet?
`d1 = {"john":40, "peter":45} d2 = {"john":466, "peter":45} d1 > d2`
- True
 - False
 - Error
 - None
26. Which of the following is a Python tuple?
- `[1, 2, 3]`
 - `(1, 2, 3)`
 - `{1, 2, 3}`
 - `{}`
27. Suppose `t = (1, 2, 4, 3)`, which of the following is incorrect?
- `print(t[3])`
 - `t[3] - 45`
 - `print(max(t))`
 - `print(len(t))`
28. What will be the output of the following Python code?
`>>>t=(1,2,4,3) >>>t[1:3]`
- `(1, 2)`
 - `(1, 2, 4)`
 - `(2, 4)`
 - `(2, 4, 3)`
29. What will be the output of the following Python code?
`>>>t=(1,2,4,3) >>>t[1:-1]`
- `(1, 2)`
 - `(1, 2, 4)`
 - `(2, 4)`
 - `(2, 4, 3)`
30. What will be the output of the following Python code? `>>>t = (1, 2, 4, 3, 8, 9)`
`>>>[t[i] for i in range(0, len(t), 2)]`
- `[2, 3, 9]`
 - `[1, 2, 4, 3, 8, 9]`
 - `[1, 4, 8]`
 - `(1, 4, 8)`