**Bakalavriat səviyyəsi üzrə**

|  |  |  |  |
| --- | --- | --- | --- |
| **Ali məktəbin adı** | | Bakı mühəndislik universiteti | |
| **İxtisasın adı** | | İnformasiya texnologiyaları və sistemləri | |
| **Fənnin adı** | | Data Stuctures | |
| **Təhsil səviyyəsi** | | Bakalavriat | |
| **Fənn bölümü (Ümumpeşə / İxtisas)** | | Ümumpeşə | |
| **Fənn bölümündə neçənci fəndir** | | 6 | |
| 1. | 3.26E4 = ? | |  | |
| A) | 0.000326 | |  | |
| B) | 32600 | | + | |
| C) | 3.30 | |  | |
| D) | 3.26 | |  | |
| E) | 3.264 | |  | |
|  |  | |  | |
| 2. | Which variable is considered to store value of 2.4563? | |  | |
| A) | int x; | |  | |
| B) | char y; | |  | |
| C) | none | |  | |
| D) | float tl; | | + | |
| E) | long a; | |  | |
|  |  | |  | |
| 3. | 1.25E-4 = ? | |  | |
| A) | 0.000125 | | + | |
| B) | 12500 | |  | |
| C) | 1.30 | |  | |
| D) | 1.256 | |  | |
| E) | 1.2 | |  | |
|  |  | |  | |
| 4. | Which variable is considered to store value of -12345? | |  | |
| A) | unsigned int x | |  | |
| B) | char y | |  | |
| C) | unsigned long z | |  | |
| D) | int t | | + | |
| E) | double d | |  | |
|  |  | |  | |
| 5. | How much information is stored in char string[10] variable? | |  | |
| A) | 1 bit | |  | |
| B) | 10 bits | |  | |
| C) | 80 bits | | + | |
| D) | 100 bits | |  | |
| E) | 800 bits | |  | |
|  |  | |  | |
| 6. | Which of the answers below can be a mantissa part of the floating point number? | |  | |
| A) | 0.3 | | + | |
| B) | 3 | |  | |
| C) | 30 | |  | |
| D) | 300 | |  | |
| E) | 3000 | |  | |
|  |  | |  | |
| 7. | Which are the types of algorithms? | |  | |
| A) | linear, branched | |  | |
| B) | ordinary, linear | |  | |
| C) | ordinary, repeated | |  | |
| D) | ordinary, branched, periodic | | + | |
| E) | none of the above | |  | |
|  |  | |  | |
| 8. | Which type of algorithm is the solution algorithm of quadratic equation? | |  | |
| A) | linear | |  | |
| B) | branched | | + | |
| C) | periodic | |  | |
| D) | complex | |  | |
| E) | c and a | |  | |
|  |  | |  | |
| 9. | What is the type of the algorithm that computes sum of N natural numbers? | |  | |
| A) | linear | |  | |
| B) | branched | |  | |
| C) | periodic | | + | |
| D) | complex | |  | |
| E) | parallel | |  | |
|  |  | |  | |
| 10. | Which of the algorithms below is an example of branched algorithm? | |  | |
| A) | computation of a factorial | |  | |
| B) | comparison of two numbers | | + | |
| C) | None | |  | |
| D) | computation of the area of a rectangle | |  | |
| E) | an approximate calculation of sin(x) function | |  | |
|  |  | |  | |
| 11. | If the address of first elemenet of float mas[10] array is 1000, what is the address of last element? | |  | |
| A) | 1000 | |  | |
| B) | 1040 | |  | |
| C) | 1036 | |  | |
| D) | 1039 | | + | |
| E) | 10 | |  | |
|  |  | |  | |
| 12. | 'say' is declared as struct type variable. Which of the following statements below is true? | |  | |
|  | struct koord {  int x;  int y;  } say; | |  | |
|  |  | |  | |
| A) | say = 25; | |  | |
| B) | x= 15; | |  | |
| C) | say\_y = 40; | |  | |
| D) | say.x = 30; | | + | |
| E) | y=12; | |  | |
|  |  | |  | |
| 13. | a = 20; | |  | |
|  | count = 1; | |  | |
|  | while (count > a) | |  | |
|  | count=count+1; | |  | |
|  | if the operations above are executed, what will be the resulting value of 'count' variable? | |  | |
| A) | 20 | |  | |
| B) | 2 | |  | |
| C) | 1 | | + | |
| D) | 19 | |  | |
| E) | 21 | |  | |
|  |  | |  | |
| 14. | In the tree tructure below, how many subbranches (children) does the root have? | |  | |
|  |  | |  | |
| A) | 2 | | + | |
| B) | 4 | |  | |
| C) | 6 | |  | |
| D) | 8 | |  | |
| E) | 9 | |  | |
|  |  | |  | |
| 15. | The tree structure is shown below. How many leaves does the tree have? | |  | |
|  |  | |  | |
| A) | 2 | |  | |
| B) | 4 | | + | |
| C) | 6 | |  | |
| D) | 8 | |  | |
| E) | 9 | |  | |
|  |  | |  | |
| 16. | Which of the structures below can be constructed dynamically? | |  | |
| A) | stack | |  | |
| B) | queue | |  | |
| C) | linked list | |  | |
| D) | all of the above | | + | |
| E) | none of the above | |  | |
|  |  | |  | |
| 17. | What is the cycle in the graphs? | |  | |
| A) | The graph in which all last elements of the routes are the same | |  | |
| B) | Endless graph | |  | |
| C) | The graph where first and last elements are the same | |  | |
| D) | The route where first and last elements are different | | + | |
| E) | None of the above | |  | |
|  |  | |  | |
| 18. | What is the traditional name of the operation where element is inserted to the stack? | |  | |
| A) | add | |  | |
| B) | insert | |  | |
| C) | append | |  | |
| D) | push | | + | |
| E) | none of the above | |  | |
|  |  | |  | |
| 19. | What is the traditional name of the operation where an element is deleted from the stack? | |  | |
| A) | delete | |  | |
| B) | peek | |  | |
| C) | pop | | + | |
| D) | remove | |  | |
| E) | none of the above | |  | |
|  |  | |  | |
| 20. | Which of the operations below can be resulted in 'stack overflow' ? | |  | |
| A) | is\_empty | |  | |
| B) | pop | |  | |
| C) | push | | + | |
| D) | init | |  | |
| E) | none of the above | |  | |
|  |  | |  | |
| 21. | Which of the problems below can be solved by using stack? | |  | |
| A) | Bracket balancing program | |  | |
| B) | Memorization of steps in labirint game | |  | |
| C) | Analysis of syntax errors for compiler | |  | |
| D) | All of the above | | + | |
| E) | None of the above | |  | |
|  |  | |  | |
| 22. | Assume that, stack with 42 elements volume has been constructed using array type. If the stack has 10 elements and these elements are placed in array data[1] - data[10], which index of the array will be used by PUSH function to insert new element? | |  | |
| A) | data[2] | |  | |
| B) | data[3] | |  | |
| C) | data[9] | |  | |
| D) | data[10] | |  | |
| E) | data[11] | | + | |
|  |  | |  | |
| 23. | One of the differences between stack and queue is: | |  | |
| A) | As compared to stacks, dynamic memory is needed for the construction of queues | |  | |
| B) | As compared to queues, dynamic memory is needed for the construction of stacks | |  | |
| C) | Queues use both ends of the structure, but stacks use just one end. | | + | |
| D) | Stacks use both ends of the structure, but queues use just one end. | |  | |
| E) | none of the above | |  | |
|  |  | |  | |
| 24. | 'A', 'L', 'I' and 'M' symbols are inserted into the queue in the given order. If they are removed from the queue one by one which of following words is obtained? | |  | |
| A) | IMLA | |  | |
| B) | MIAL | |  | |
| C) | ALMI | |  | |
| D) | LAIM | |  | |
| E) | None of the above | | + | |
|  |  | |  | |
| 25. | If we construct a queue like a linked list, where will the new element be inserted in the this list? | |  | |
| A) | Beginning | |  | |
| B) | End | | + | |
| C) | After all elements those are bigger than the new element | |  | |
| D) | After all elements those are smaller than the new element | |  | |
| E) | None of the above | |  | |
|  |  | |  | |
| 26. | Queue is constructed like a linked list and two pointers are pointing to the beginning and end of the queue. If the new element is inserted to non-empty queue, which of these two pointers change? | |  | |
| A) | None of the pointers change. | |  | |
| B) | The pointer which points to the beginning of queue changes. | |  | |
| C) | The pointer which points to the end of queue changes. | | + | |
| D) | Both of them. | |  | |
| E) | None of the answers above is true. | |  | |
|  |  | |  | |
| 27. | Queue is constructed like a linked list and two pointers are pointing to the beginning and end of the queue. If the new element is inserted to empty queue, which of these two pointers change? | |  | |
| A) | None of the pointers change. | |  | |
| B) | The pointer which points to the beginning of queue changes. | |  | |
| C) | The pointer which points to the end of queue changes. | | + | |
| D) | Both of them. | |  | |
| E) | None of the answers above is true. | |  | |
|  |  | |  | |
| 28. | Assume that, the pointer named 'cursor' points to one element in a linked list. Elements of the linked list has 'data' and 'link' fields. Which of the expressions below has to be used in order to make 'cursor' point to the next element in the linked list? | |  | |
| A) | cursor:=cursor+1; | |  | |
| B) | cursor^: = link; | |  | |
| C) | cursor: = link; | |  | |
| D) | cursor = cursor^.link; | | + | |
| E) | none of the above | |  | |
|  |  | |  | |
| 29. | Assume that, the pointer named 'cursor' points to one element in a linked list. If 'cursor' points to a last element of of this list, which of the expressions below is true? | |  | |
| A) | cursor = NIL | |  | |
| B) | cursor^.link = NIL | | + | |
| C) | cursor^.data = NIL | |  | |
| D) | cursor^.data = 0.0 | |  | |
| E) | None of the above | |  | |
|  |  | |  | |
| 30. | In the image, how many elements of tree structure has at least one sibling element? | |  | |
|  |  | |  | |
| A) | 5 | |  | |
| B) | 6 | | + | |
| C) | 7 | |  | |
| D) | 8 | |  | |
| E) | 9 | |  | |
|  |  | |  | |
| 31. | In the tree structure shown in the picture, which value is kept by the parent of the element that keeps the value 30? | |  | |
|  |  | |  | |
| A) | 10 | |  | |
| B) | 11 | | + | |
| C) | 14 | |  | |
| D) | 40 | |  | |
| E) | None of the above | |  | |
|  |  | |  | |
| 32. | In the tree structure shown in the picture, how many descendants does the root element have? | |  | |
|  |  | |  | |
| A) | 0 | |  | |
| B) | 3 | |  | |
| C) | 4 | |  | |
| D) | 8 | | + | |
| E) | 9 | |  | |
|  |  | |  | |
| 33. | What is depth of the tree structure shown in the picture? | |  | |
|  |  | |  | |
| A) | 2 | |  | |
| B) | 3 | | + | |
| C) | 4 | |  | |
| D) | 8 | |  | |
| E) | 9 | |  | |
|  |  | |  | |
| 34. | T is a binary tree with 14 elements. What is the possible minimum depth of T? | |  | |
| A) | 0 | |  | |
| B) | 3 | | + | |
| C) | 4 | |  | |
| D) | 5 | |  | |
| E) | none of the above | |  | |
|  |  | |  | |
| 35. | In the tree structure shown in the picture, what is the sequence of elements after preorder traversal? | |  | |
|  | 12 | |  | |
|  | / \ | |  | |
|  | 2 14 | |  | |
|  | / \ \ | |  | |
|  | 1 3 30 | |  | |
|  | \ \ | |  | |
|  | 7 40 | |  | |
| A) | 1-2-3-7-14-12-30-40 | |  | |
| B) | 1-2-3-14-7-12-40-30 | |  | |
| C) | 1-3-2-7-40-30-14-12 | |  | |
| D) | 12-2-1-3-7-14-30-40 | |  | |
| E) | none of the above | |  | |
|  |  | |  | |
| 36. | In the tree structure shown in the picture, what is the sequence of elements after in order traversal? | |  | |
|  | 12 | |  | |
|  | / \ | |  | |
|  | 2 14 | |  | |
|  | / \ \ | |  | |
|  | 1 3 30 | |  | |
|  | \ \ | |  | |
|  | 7 40 | |  | |
| A) | 1-2-3-7-12-14-30-40 | | + | |
| B) | 1-2-3-12-7--14-40-30 | |  | |
| C) | 1-3-2-7-12-40-30-14 | |  | |
| D) | 14-2-1-3-12-7-30-40 | |  | |
| E) | none of the above | |  | |
|  |  | |  | |
| 37. | In the tree structure shown in the picture, what is the sequence of elements after postorder traversal? | |  | |
|  |  | |  | |
|  | 12 | |  | |
|  | / \ | |  | |
|  | 2 14 | |  | |
|  | / \ \ | |  | |
|  | 1 3 30 | |  | |
|  | \ \ | |  | |
|  | 7 40 | |  | |
|  |  | |  | |
| A) | 1-7-3-2-40-30-12-14 | |  | |
| B) | 1-3-7-2-40-30-12 | |  | |
| C) | 3-7-1-2-30-40-12 | |  | |
| D) | 12-14-30-40-2-3-7-1 | |  | |
| E) | none of the above | | + | |
|  |  | |  | |
| 38. | Which of the structures given below is constructed using FILO (first in, last out) logic? | |  | |
| A) | array | |  | |
| B) | stack | | + | |
| C) | queue | |  | |
| D) | linked list | |  | |
| E) | binary tree | |  | |
|  |  | |  | |
| 39. | Which of the structures given below is constructed using FIFO (first in, first out) logic? | |  | |
| A) a | array | |  | |
| B) s | stack | |  | |
| C) | queue | | + | |
| D) l | linked list | |  | |
| E) b | binary tree | |  | |
|  |  | |  | |
| 40. | Which method is used to define the index of location of the data by using special "key" function? | |  | |
| A) | array use | |  | |
| B) | hashing | | + | |
| C) | cashing | |  | |
| D) | sorting | |  | |
| E) | none of the above | |  | |
|  |  | |  | |
| 41. | If hash function defines the same hash index for two different data, what hashing problem is this? | |  | |
| A) | collision | | + | |
| B) | revision | |  | |
| C) | encapsulation | |  | |
| D) | modulation | |  | |
| E) | none | |  | |
|  |  | |  | |
| 42. | If the sorting operation is done by the given principle below, which sorting algorithm is used for this purpose? | |  | |
|  | 1. 28 18 7 4 10 | |  | |
|  | 1. 4 18 7 28 10 | |  | |
|  | 1. 4 7 18 28 10 | |  | |
|  | 1. 4 7 10 28 18 | |  | |
|  | 1. 4 7 10 18 28 | |  | |
| A) | bubble sort | |  | |
| B) | selection sort | | + | |
| C) | insertion sort | |  | |
| D) | merge sort | |  | |
| E) | none of the above | |  | |
|  |  | |  | |
| 43. | This sorting algorithm compares each element with the previous element starting from the last and if needed swaps their position. The algorithm stops in the case where none of elements are swapped with others. Which algorithm is this? | |  | |
| A) | bubble sort | | + | |
| B) | selection sort | |  | |
| C) | insertion sort | |  | |
| D) | merge sort | |  | |
| E) | none of the above | |  | |
|  |  | |  | |
| 44. | Algorithm used – insertion sort. Write the result of 4-th line: | |  | |
|  | 1. 28 18 7 4 10 | |  | |
|  | 1. 18 28 7 4 10 | |  | |
|  | 1. 7 18 28 4 10 | |  | |
|  | 1. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |  | |
| A) | 7-18-28-10-4 | |  | |
| B) | 10-7-18-28-4 | |  | |
| C) | 4-7-18-28-10 | | + | |
| D) | 4-18-28-7-10 | |  | |
| E) | none of the above | |  | |
|  |  | |  | |
| 45. | Which sorting algorithm is performed by the procedure below? | |  | |
|  | procedure Sort MyArray; | |  | |
|  | var i, j, tmp: integer; | |  | |
|  | begin | |  | |
|  | for r i:=size – 1 downto 1 do | |  | |
|  | for j:= 1 to i do | |  | |
|  | if MyArray[j] > MyArray[j+1] then | |  | |
|  | begin | |  | |
|  | Tmp:= MyArray[j]; | |  | |
|  | MyArray[j]:= MyArray[j+1]; | |  | |
|  | MyArray [j+1]:=tmp; | |  | |
|  | end; | |  | |
|  | end; | |  | |
| A) | bubble sort | | + | |
| B) | selection sort | |  | |
| C) | insertion sort | |  | |
| D) | merge sort | |  | |
| E) | none of the above | |  | |
|  |  | |  | |
| 46. | Which operation is performed by the procedure written in the pseudocode below? | |  | |
|  | Put the root node on a stack; | |  | |
|  | while (stack is not empty) { | |  | |
|  | remove a node from the stack; | |  | |
|  | if (node is a goal node) return success; | |  | |
|  | put all children of the node onto the stack; | |  | |
|  | } | |  | |
|  | return failure; | |  | |
| A) | Sorts stack elements in ascending order | |  | |
| B) | Implements Breadth-First search in tree structure. | |  | |
| C) | Implements Breadth-First search in graph structure. | |  | |
| D) | Implements Depth-First search in tree structure. | | + | |
| E) | Implements Depth-First search in graph structure. | |  | |
|  |  | |  | |
| 47. | What is "path" in graphs? | |  | |
| A) | Path is different name for cycle in the graph. | |  | |
| B) | Path denotes number of elements of the graph. | |  | |
| C) | Path is the sequence of edges where sequentially each graph element is a child of next element. | | + | |
| D) | All expressions above are true. | |  | |
| E) | All expressions above are false. | |  | |
|  |  | |  | |
| 48. | What is the purpose of using Dijkstra algorithm? | |  | |
| A) | Dijkstra algorithm is used to construct acyclic graphs. | |  | |
| B) | Dijkstra algorithm finds shortest route from the given element to other elements of the graph. | | + | |
| C) | Dijkstra algorithm arranges all elements of the graph in descending order. | |  | |
| D) | Dijkstra algorithm arranges all elements of the graph in ascending order. | |  | |
| E) | none of the above | |  | |
|  |  | |  | |
| 49. | Which operation is performed by the procedure written in the pseudocode below? | |  | |
|  | Put the starting node on a stack; | |  | |
|  | While (stack is not empty) { | |  | |
|  | remove a node from the stack; | |  | |
|  | if (node is a goal node) return success; | |  | |
|  | put all unvisited adjacent nodes of the node onto the stack; | |  | |
|  | } | |  | |
|  | return failure; | |  | |
| A) | Sorts stack elements in ascending order | |  | |
| B) | Implements Breadth-First search in tree structure. | |  | |
| C) | Implements Breadth-First search in graph structure. | |  | |
| D) | Implements Depth-First search in tree structure. | |  | |
| E) | Implements Depth-First search in graph structure. | | + | |
|  |  | |  | |
| 50. | Which operation is performed by the procedure written in the pseudocode below? | |  | |
|  | Put the starting node in the queue; | |  | |
|  | While (queue is not empty) { | |  | |
|  | Remove a node from the beginning of the queue; | |  | |
|  | If (node is a goal node) return success; | |  | |
|  | put all unvisited adjacent nodes of the node into the queue; | |  | |
|  | } | |  | |
|  | return failure; | |  | |
| A) | Sorts stack elements in ascending order | |  | |
| B) | Implements Breadth-First search in tree structure. | |  | |
| C) | Implements Breadth-First search in graph structure. | | + | |
| D) | Implements Depth-First search in tree structure. | |  | |
| E) | Implements Depth-First search in graph structure. | |  | |
|  |  | |  | |
| 51. | Which operation is performed by the procedure written in the pseudocode below? | |  | |
|  | Put the root node in the queue; | |  | |
|  | while (queue is not empty) { | |  | |
|  | remove a node from the beginning of the queue; | |  | |
|  | if (node is a goal node) return success; | |  | |
|  | put all children of the node into the queue; | |  | |
|  | } | |  | |
|  | return failure; | |  | |
| A) | Sorts stack elements in ascending order. | |  | |
| B) | Implements Breadth-First search in tree structure. | | + | |
| C) | Implements Breadth-First search in graph structure. | |  | |
| D) | Implements Depth-First search in tree structure. | |  | |
| E) | Implements Depth-First search in graph structure. | |  | |
|  |  | |  | |
| 52. | Compute the complexity of the given part of a program. | |  | |
|  | for(i=0; i<n; i++)  for (j=i; j<n; j++)  sum = sum + a[j]; | |  | |
| A) | O(log n) | |  | |
| B) | O(n) | |  | |
| C) | O(n2) | | + | |
| D) | O(nlog n) | |  | |
| E) | O(1) | |  | |
|  |  | |  | |
| 53. | Compute the complexity of the given part of a program. | |  | |
|  | for (i=0; i<n; i++) | |  | |
|  | if(odd(i) | |  | |
|  | { | |  | |
|  | for (j=0; j<n; j++) | |  | |
|  | sum = sum + a[j]; | |  | |
|  | } | |  | |
|  | else | |  | |
|  | sum = sum + a[i]; | |  | |
|  |  | |  | |
| A) | O(log n) | |  | |
| B) | O(n2) | | + | |
| C) | O(n) | |  | |
| D) | O(nlog n) | |  | |
| E) | O(1) | |  | |
|  |  | |  | |
| 54. | Compute the complexity of the given part of a program. | |  | |
|  | for (i=0; i<n; i++)  for (j=0; j<n; j++)  sum = sum + a[j]; | |  | |
| A) | O(log n) | |  | |
| B) | O(n) | |  | |
| C) | O(nlog n) | |  | |
| D) | O(n2) | | + | |
| E) | O(1) | |  | |
|  |  | |  | |
| 55. | Compute the complexity of the given part of a program. | |  | |
|  | for (i=0; i<n; i++)  sum = sum + a[i]; | |  | |
| A) | O(log n) | |  | |
| B) | O(n) | | + | |
| C) | O(n2) | |  | |
| D) | O(nlog n) | |  | |
| E) | O(1) | |  | |
|  |  | |  | |
| 56. | The array A has n number of elements. Compute the complexity of the algorithm that finds m smallest elements of this array. (m is much smaller than n). | |  | |
| A) | O(m) | |  | |
| B) | O(n) | |  | |
| C) | O(n2) | |  | |
| D) | O(nlog n) | |  | |
| E) | O(mn) | | + | |
|  |  | |  | |
| 57. | Find the result of the given prefix expression. | |  | |
|  | / –\* 2 5 \* 1 2 – 11 9 | |  | |
| A) | 25 | |  | |
| B) | 11 | |  | |
| C) | 5 | |  | |
| D) | 4 | | + | |
| E) | 9 | |  | |
|  |  | |  | |
| 58. | Find the result of the given prefix expression. | |  | |
|  | 1 6 \* 2 2\* - 10 8 - / | |  | |
| A) | 16 | |  | |
| B) | 1 | | + | |
| C) | 10 | |  | |
| D) | 8 | |  | |
| E) | 22 | |  | |
|  |  | |  | |
| 59. | Which is the postfix equivalent of the given infix expression? | |  | |
|  | A+B^(C+D)/E | |  | |
| A) | A+B^(C+D)/E | |  | |
| B) | ABCD+^E/+ | | + | |
| C) | +A/^B+CDE | |  | |
| D) | AB+^C+D/E | |  | |
| E) | +AB^C+/DE | |  | |
|  |  | |  | |
| 60. | Which is the postfix equivalent of the given infix expression? | |  | |
|  | (A+B)^C-D\*(E+F) | |  | |
| A) | -^+ABC\*D+EF | |  | |
| B) | A+B^C-D\*E+F | |  | |
| C) | ABC^DEF+\*- | | + | |
| D) | (A+B)^C-D\*(E+F) | |  | |
| E) | ABCDEF+^-\*+ | |  | |
|  |  | |  | |
| 61. | Which is the prefix equivalent of the given infix expression? | |  | |
|  | (A+B)\*(C+D)-E^F-G | |  | |
| A) | A+B\*C+D-E^F-G | |  | |
| B) | (A+B)\*(C+D)-E^F-G | |  | |
| C) | AB+CD+\*EF^-G- | |  | |
| D) | +\*+-^-ABCDEFG | |  | |
| E) | --\*+AB+CD^EFG | | + | |
|  |  | |  | |
| 62. | Which is the prefix equivalent of the given infix expression? | |  | |
|  | (A+B)/(C-D)^E+F | |  | |
| A) | )+/+AB^-CDEF | | + | |
| B) | ABCDEF+/-^+ | |  | |
| C) | (A+B)/(C-D)^E+F | |  | |
| D) | A+B/C-D^E+F | |  | |
| E) | +/-^+ABCDEF | |  | |
|  |  | |  | |
| 63. | The data structure is required that can only store (unique) elements and this structure has to support adding and removing elements. Which is the appropriate data structure that can be used for this purpose? | |  | |
| A) | list | |  | |
| B) | queue | |  | |
| C) | Set | | + | |
| D) | hash table | |  | |
| E) | stack | |  | |
|  |  | |  | |
| 64. | Data structure is: | |  | |
| A) | Set of principles and restrictions which defines relation between different groups and elements of data | | + | |
| B) | Set of principles and restrictions which defines relationship between different elements of data | |  | |
| C) | Set of principles and restrictions which defines relationship between different groups of data | |  | |
| D) | arbitrary hierarchical data | |  | |
| E) | independent data sets | |  | |
|  |  | |  | |
| 65. | Which is the linear structure that gives us a chance to only obtain the last element? | |  | |
| A) | Stack | | + | |
| B) | queue | |  | |
| C) | deck | |  | |
| D) | array | |  | |
| E) | list | |  | |
|  |  | |  | |
| 66. | Which data structure supports FIFO (First In First out) principle of operations on its elements? | |  | |
| A) | stack | |  | |
| B) | deck | |  | |
| C) | queue | | + | |
| D) | list | |  | |
| E) | tree | |  | |
|  |  | |  | |
| 67. | Which is the linear sequential list that the elements can imported and exported from the front end and the back end? | |  | |
| A) | stack | |  | |
| B) | queue | |  | |
| C) | Deck | | + | |
| D) | linked list | |  | |
| E) | tree | |  | |
|  |  | |  | |
| 68. | What is the primary feature of the queue? | |  | |
| A) | Open from the both sides | | + | |
| B) | Open from one side for insertion and deletion operations | |  | |
| C) | Arbitrary element is accessible | |  | |
| D) | Elements can not be deleted | |  | |
| E) | Only insertion operations are supported | |  | |
|  |  | |  | |
| 69. | What is the primary feature of the stack? | |  | |
| A) | Open from the both sides for insertion and deletion operations | |  | |
| B) | Arbitrary element is accessible | |  | |
| C) | Only insertion operations are supported | |  | |
| D) | Open from one side for insertion and deletion operations | | + | |
| E) | Elements can not be deleted | |  | |
|  |  | |  | |
| 70. | Which data structure supports LIFO (Last In First Out) principle of operations on its elements? | |  | |
| A) | Stack | | + | |
| B) | Deck | |  | |
| C) | Queue | |  | |
| D) | LIst | |  | |
| E) | Tree | |  | |
|  |  | |  | |
| 71. | Which operation reads the top element of the stack without removing it? | |  | |
| A) | pop; | |  | |
| B) | push; | |  | |
| C) | top | | + | |
| D) | empty | |  | |
| E) | size | |  | |
|  |  | |  | |
| 72. | Which is the element selection principle from the stack? | |  | |
| A) | first element | |  | |
| B) | last element | | + | |
| C) | arbitrary element | |  | |
| D) | first and last element | |  | |
| E) | central element | |  | |
|  |  | |  | |
| 73. | It is appropriate to describe a binary tree in a computer memory using the following form: | |  | |
| A) | linked linear list | |  | |
| B) | array | |  | |
| C) | linked non-linear list | | + | |
| D) | stack | |  | |
| E) | deck | |  | |
|  |  | |  | |
| 74. | Tree is called a binary tree when degree of nodes is equal to: | |  | |
| A) | 2 or 0; | | + | |
| B) | yalnız 2; | |  | |
| C) | M or 0; | |  | |
| D) | M | |  | |
| E) | arbitrary number | |  | |
|  |  | |  | |
| 75. | Which of the efficient sorting criteria is defined as M=0.01\*n\*n+10\*n? | |  | |
| A) | number of comparisons | | + | |
| B) | time spent for writing a program | |  | |
| C) | number of displacements | |  | |
| D) | time spent for sorting | |  | |
| E) | number of iterations | |  | |
|  |  | |  | |
| 76. | Which is sorting that is done in RAM? | |  | |
| A) s | sorting of address tables | |  | |
| B) c | complete sorting | |  | |
| C) s | sorting with direct insertion | |  | |
| D) | inner sorting | | + | |
| E) | outer sorting | |  | |
|  |  | |  | |
| 77. | How to decrease machine time when sorting big volume of data? | |  | |
| A) | Doing sorting operation in key addresses table | | + | |
| B) | Doing sorting operation in more powerful computer | |  | |
| C) | Doing sorting operation by dividing data to smaller parts | |  | |
| D) | By decreasing number of examples in the data | |  | |
| E) | Doing sorting operation in many computers together | |  | |
|  |  | |  | |
| 78. | Sorting process is called stable, if in the sorting operation ... | |  | |
| A) | The relative location of the elements is not important | |  | |
| B) | The relative location of the elements which have the same key do not change | | + | |
| C) | The relative location of the elements which have the same key changes | |  | |
| D) | The relative location of the elements is not defined | |  | |
| E) | The number of elements is not important | |  | |
|  |  | |  | |
| 79. | Which of the notions defined below is one of the sorting types? | |  | |
| A) | inner sorting | | + | |
| B) | descending sort | |  | |
| C) | sorting of input | |  | |
| D) | ascending sort | |  | |
| E) | relative sorting | |  | |
|  |  | |  | |
| 80. | How many comparison operations does highly efficient sorting require? | |  | |
| A) | n\*log(n) | |  | |
| B) | n; | |  | |
| C) | n\*n/4 | |  | |
| D) | n2 | |  | |
| E) | 2n | |  | |
|  |  | |  | |
| 81. | How many comparison and shifting operations of the elements are required in bubble sort? | |  | |
| A) n | n\*logn(n) | |  | |
| B) | n2/4 | | + | |
| C) | (n2-n)/2 | |  | |
| D) | n2 | |  | |
| E) | 2n | |  | |
|  |  | |  | |
| 82. | What is the idea behind QuickSort (fast sorting) method? | |  | |
| A) | Choosing the 1, 2, ..., ..., n-th elements for comparison with others? | |  | |
| B) | Making keys for elements which has been chosen | | + | |
| C) | Changing the location of neighboring elements | |  | |
| D) | The comparison of not neighboring elements in pairs | |  | |
| E) | Usage of additional array | |  | |
|  |  | |  | |
| 83. | If we traversing from left to right in the tree which sequence we get ... | |  | |
| A) | sorting based on descending order | |  | |
| B) | not sorted | | + | |
| C) | sorting based on ascending order | |  | |
| D) | sorting based on partly descending, partly ascending order | |  | |
| E) | sorting based on partly ascending, partly descending order | |  | |
|  |  | |  | |
| 84. | Where the linear search is efficient? | |  | |
| A) | in the list | |  | |
| B) | in the array | |  | |
| C) | in the array and list | | + | |
| D) | in the stack | |  | |
| E) | in the queue | |  | |
|  |  | |  | |
| 85. | Which searching method is more efficient? | |  | |
| A) | linear | |  | |
| B) | Binary | | + | |
| C) | by insertion | |  | |
| D) | using obstacles | |  | |
| E) | by shifting | |  | |
|  |  | |  | |
| 86. | How the elements of the array are placed in binary search? | |  | |
| A) | in asceding order | | + | |
| B) | in arbitrary order | |  | |
| C) | in descending order | |  | |
| D) | in partly ascending, and partly descending order | |  | |
| E) | in partly descending, and partly ascending order | |  | |
|  |  | |  | |
| 87. | What is the idea behind the linear search? | |  | |
| A) | examination is done over two elements staring from beginning to the end | |  | |
| B) | elements are examined sequentially starting from center of the list | |  | |
| C) | each element is examined sequentially | | + | |
| D) | each time examination is done over elements by dividing the list into two parts | |  | |
| E) | each element is examined sequentially after the list is sorted | |  | |
|  |  | |  | |
| 88. | Which element does not reference to the other elements of the tree? | |  | |
| A) | root | |  | |
| B) | leaf | | + | |
| C) | node | |  | |
| D) | parent | |  | |
| E) | branch | |  | |
|  |  | |  | |
| 89. | Which element is not referenced by the other elements of the tree? | |  | |
| A) | Root | | + | |
| B) | leaf | |  | |
| C) | node | |  | |
| D) | child | |  | |
| E) | grandchild | |  | |
|  |  | |  | |
| 90. | The depth of the tree is called ... | |  | |
| A) | maximum number of nodes | |  | |
| B) | maximum number of relations | |  | |
| C) | maximum number of leaves | |  | |
| D) | maximum length of the route from the root to the leaf | | + | |
| E) | maximum number of the leaves | |  | |
|  |  | |  | |
| 91. | The degree of the tree is called ... | |  | |
| A) | the maximum degree of all nodes | | + | |
| B) | the maximum number of levels of its nodes | |  | |
| C) | the maximum number nodes | |  | |
| D) | the maximum number of relations | |  | |
| E) | the maximum number of leaves | |  | |
|  |  | |  | |
| 92. | How the length of tree route is defined? | |  | |
| A) | by number of edges from the node to the root | |  | |
| B) | by the number of edges from leaves to the root | |  | |
| C) | by the maximum number of edges | |  | |
| D) | by the maximum number of leaves | |  | |
| E) | by the longest route from the root to the lowest level leaf | | + | |
|  |  | |  | |
| 93. | The tree is called binary tree when ... | |  | |
| A) | the number of nodes is zero, or it embraces root of other two subtrees | | + | |
| B) | each node has at least two ancestors | |  | |
| C) | the number of levels from the root to the leaf is at most two | |  | |
| D) | the number of levels from the root to the leaf is at least two | |  | |
| E) | each node has more than two subnodes | |  | |
|  |  | |  | |
| 94. | Vertices of the graph can be corresponded with ... | |  | |
| A) | interaction between objects | |  | |
| B) | objects | | + | |
| C) | relations | |  | |
| D) | types of interactions | |  | |
| E) | sets | |  | |
|  |  | |  | |
| 95. | Edges of the graph can be corresponded with ... | |  | |
| A) | the relations between sets | |  | |
| B) | the types of interactions | |  | |
| C) | sets | |  | |
| D) | objects | |  | |
| E) | the relations between objects | | + | |
|  |  | |  | |
| 96. | The graph which only has edges is called: | |  | |
| A) | directed | |  | |
| B) | undirected | | + | |
| C) | ordinary | |  | |
| D) | mixed | |  | |
| E) | complex | |  | |
|  |  | |  | |
| 97. | Which of the algorithms below cannot be solved recursively? | |  | |
| A) | computing sum of the numbers from 1 to N | |  | |
| B) | computing N! | |  | |
| C) | computing xn | |  | |
| D) | solving quadratic equation | | + | |
| E) | checking whether a is divided by b without remainder | |  | |
|  |  | |  | |
| 98. | Which of the algorithms below can be solved recursively? | |  | |
| A) | summing up a and b | |  | |
| B) | computing N! | | + | |
| C) | comparison of three numbers | |  | |
| D) | finding the area of a triangle | |  | |
| E) | comparison of two lines | |  | |
|  |  | |  | |
| 99. | In the recursive implementation of N! , what is the condition to terminate the loop? | |  | |
| A) | if(n > 0) retunr n\*factor(n-1); | |  | |
| B) | if(n) return 0; | |  | |
| C) | if(n > 0) return n; | |  | |
| D) | if(n == 0) return 1; | | + | |
| E) | f(n > 0) return 1; | |  | |
|  |  | |  | |
| 100. | Which line describes the recursive implementation xn correctly? | |  | |
| A) | return x\*pow(x-1, n-1); | |  | |
| B) | return x\*n; | |  | |
| C) | return x\*pow(x-1, n); | |  | |
| D) | return x\*pow(x, n+1); | |  | |
| E) | return x\*pow(x, n-1); | | + | |