IDEAOA02

- Which is not a property of a algorithm?

Select one:

- It must terminate for all inputs.
- The order of the algorithm's steps must be precisely defined.
- The execution time and the memory needed for a algorithm must be percisely defined.
- It must be correct and composed of precisely defined steps.

IDEAOA03

– Which statement below is wrong?

Select one:

- For the same data, some data structures may require more or less space.
- A data structure is a way of organizing data for processing within a computer program.
- A data structure is a piece of information (a physical instantiation of a data type)
- For the same operations on the data, some data structures lead to more or less efficient algorithms.

IDEAOA04

– Which statement is correct concerning to the complexity of algorithm?

Select one:

- The complexity of an algorithm is determined by the total lines of code of the program that implements the algorithm using a given programming language.
- The complexity of an algorithm is a measure of the amount of time and cost needed to implement this algorithm.
- The complexity of an algorithm is a measure of the amount of time and space required by the algorithm for an input of a given size n.
- The complexity of an algorithm is determined by the maximum value of the input size n that does not affect the correctness of the algorithm.

IDEAOA05

– When evaluating algorithm's complexity, which approach makes possible an evaluation that is independent of the hardware and software environments?

Select one:

- Using input data sets of varying size.
- Theoretical approach.
- Measuring the running time and memory space using the same hardware and software environment.
- Experimental approach.

IDEAOA06

– What is time complexity of an algorithm?

Select one:

- The amount of time needed to implement the algorithm.
- The amount of time that the algorithm needs to run for an input of a given size n.
- The upper limits for excution time of the algorithm.
- The response time of the algorithm.

IDEAOA08

– Which statement is wrong concerning to the best-case time complexity of an algorithm?

Select one:

- The best case of an algorithm A is estimated as the minimum number of primitive operations performed by A on an input size n.
- Many algorithms perform exactly the same in the best case.
- The best-case is used frequently to analyze the time complexity of algorithms.
- The best-case gives us an lower bound on the time complexity of algorithms.

IDEAOA09

– Which statement is wrong concerning to the average-case time complexity of an algorithm?

Select one:

- The average-case is places somewhere between the best-case and the worse-case.
- The average-case of an algorithm A is estimated as the average number of primitive operations performed by A on an input size n.

- The average-case of an algorithm A is depended on the characteristic of the input data.
- The average-case is easy to determine.

IDEAOA10

– Which statement is wrong concerning to the worst-case time complexity of an algorithm?

Select one:

- The worst case gives us an upper bound on time complexity of an algorithm.
- The worst-case of an algorithm A is estimated as the maximize number of primitive operations performed by A on an input size n.
- At the worst-case the algorithm takes more time to finish than it does at the average-case and best-case.
- The worst-case is not very informative because many algorithms rarely perform at their worst-case.

IDEAOA11

– Which one determines the asymptotic behavior of the function T(n)?

Select one:

- The leading term.
- The term has the biggest coefficient.
- The first term.
- The last term.

IDEAOA12

– Which notation represents the upper-bound of the grow rate of a function?

Select one:

- Big-Theta notation
- Big-Omega notation
- Big-Oh notation
- Big-Alpha notation

IDEAOA14

– Suppose that the estimated time complexity of algorithm A and algorithm B is TA(N) and TB(N) respectively. How can we compare the time complexity of A and B?

Select one:

- We compare the value of TA and TB corresponding to some special value of n.

- We compare the value of TA and TB corresponding to a very large, pre-defined value of n.
- We compare the value of TA and TB corresponding to every value of n.
- We compare the grow rate of the leading terms of TA(N) and TB(N).