



Analysis of Algorithms
CS 312 - HBD1
Department of Physics and Computer Science
Medgar Evers College
Exam 1

Instructions:

- The exam requires completing a set of tasks within 80 minutes.
- Write your solutions in the blue book provided.
- Use the pseudocode guidelines for Question 1.
- Case Study: In business, the median, which represents the middle value in an ordered data set [the middle value for odd count sets and the average of middle values for even count sets], provides a balanced and reliable indicator of the central value in data sets that may contain outliers or skewed distributions, making it essential for accurate decision-making, fair analysis, and realistic planning.
Your objective is to develop a median algorithm, and then analyze and test it.
- Cheating of any kind is prohibited and will not be tolerated.
- Violating and/or failing to follow any of the rules will result in an automatic zero (0) for the exam.

TO ACKNOWLEDGE THAT YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS ABOVE,
PRINT YOUR NAME AND THE DATE ON YOUR SUBMISSIONS

Grading

Section	Maximum Points	Points Earned
1	5	
2	5	
3	4	
4	2	
5	4	
Total	20	

1. Define an algorithm (in pseudocode) named `Median()` that takes an unsorted number array as input and returns the median value of the array.
2. Construct a runtime table and determine the runtime function for the `Median()` algorithm you defined in Question 1. Before constructing the table, clearly specify the input size. The runtime analysis should be performed for the worst-case scenario, assuming that the cost of each operation is 1.
3. Simulate the execution of the `Median()` algorithm for each of the input examples provided. Each simulation should be a detailed record list of any changes made to the input data and the significant data structures utilized by the algorithm, ending with the resulting output.
 - [3,2,8,5,1,7,4,6]
 - [6,3,2,9,4,5,7,1,8]
4. State and prove the Θ -notation asymptotic bound for the `Median()` algorithm described in Question 1. The bound must be expressed as a parent function (a monomial with no coefficient or 1). You must show work to receive credit.
5. Create a C++ program that defines a function implementing the `Median()` algorithm you described in Question 1. The program should test this function by comparing its output to the average (mean) of the same array input and displaying both results. The function header should include any additional parameters necessary for its correct execution.

Pseudocode Guidelines

The class pseudocode will be presented as a text-friendly numbered list. It will contain all elements of the procedural and object-oriented programming paradigms; however, it eliminates data abstraction, modularity, and error handling. In general, the pseudocode conventions are

- Every statement represents a simple operation and is numbered.
- Indentation will indicate a block structure (body statements). Each block structure begins its numbering from 1. Once a block structure is exited, the numbering continues from where it was before entering the block structure.
- The assignment operator is `<-`.
- The arithmetic operators are the standard operators (`+`, `-`, `*`, `/`, `%`).
- The logical operators are **and**, **or**, and **not**.
- The relational operators are `=`, `!=`, `<`, `>`, `<=`, `>=`.
- The selection statement syntaxes are

1. `if condition, then body`
2. `if condition, then body else body`
3. `if condition, then body elif condition, then body`

where syntax 3 can contain multiple `elif` statements after `if` statement and can end with a `else` statement.

- The loop statement syntaxes are

1. `while condition, do body`
2. `for initialization to end by step, do body`
3. `foreach variable in sequence, do body`

where syntax 2 can omit the `by` clause which means steps are 1. Furthermore, loop interrupt statements `break` and `continue` are available.

- Variables are local. They are global only if they begin with an underscore.
- Arrays can access their elements with the subscript operator (`[]`) as expected with indices ranging from `[0,n)` where `n` is the size of the array. Additionally, arrays have the attribute `length` that is accessed with the dot operator (`.`) as follows

Example:

`A.length` provides the size of the array `A`

- Objects contain both fields and methods, which are accessed using the dot operator. They behave as pointers; that is, assigning an object to another makes the objects aliases. Likewise, an object can be assigned `nil`.
- All function parameters are passed by value, except for arrays and objects that are passed by reference. To explicitly make a parameter pass by reference, precede its name with an ampersand in the function header. Additionally, a function can return 0 or more outputs.