COMPUTER APPLICATIONS (JAVA)

Class 10 | Chapter: Arrays

Question Bank: Linear Search, Binary Search, Selection Sort and Bubble Sort

Name: _____ Section: _____ Date: _____

Instructions

- Total questions: 50 programming questions (10 each on Linear Search, Binary Search, Selection Sort and Bubble Sort, followed by 10 mixed questions). Marks are shown in brackets.
- Write complete Java programs using the Scanner class for input. Include a variable description table or comments.
- Array elements may be integers, characters or Strings; use the String methods equals(), equalsIgnoreCase(), compareTo() and compareToIgnoreCase() appropriately.
- Position means index + 1 unless the question says otherwise. Use only the sorting or searching technique named in each question (no Arrays.sort()).

Section A: Linear Search

Q1. Write a program to accept 10 integers in an array and a number to be searched. Using linear search, display the position (index + 1) at which the number is found, or the message "Not Found". **[5]**

Q2. Write a program to store the names of 8 cities in a String array. Accept a city name from the user and search for it using linear search, ignoring the case of the letters. Display "Found at position __" or "Not Found". **[5]**

Q3. Write a program to accept 15 integers in an array and a number to be searched. Using linear search, display how many times the number occurs along with all the indices at which it occurs. If it does not occur, display "Not Found". **[6]**

```
Sample:
Number to search: 7
Occurs 3 time(s) at index: 2 6 11
```

Q4. Write a program to accept a word, convert it to uppercase and store its letters in a character array using toCharArray(). Accept a character from the user and, using linear search, display the index of its first occurrence and of its last occurrence in the word. If it is absent, display "Character not present". **[6]**

Q5. Write a program to store 10 words in a String array. Accept a letter from the user and, using linear search, display all the words that begin with that letter (ignore case) and also display how many such words were found. **[6]**

Q6. The marks (out of 100, can be decimal) of 10 students are stored in a double array, where the roll numbers are 1 to 10 in order. Accept a mark from the user and, using linear search, display the roll numbers of all students who obtained exactly that mark. **[6]**

Q7. Write a program to store 6 names in a String array. Define a user-defined method int linearSearch(String a[], String key) that returns the index of the key if found and -1 otherwise. Call it from main() and display a suitable message. **[6]**

Q8. Write a program to accept a sentence. Store the vowels 'a', 'e', 'i', 'o', 'u' in a character array. Using linear search on this array, check every character of the sentence (ignore case) and display the total number of vowels, consonants and other characters (spaces, digits, symbols). **[6]**

Q9. A school stores the roll numbers in an int array, the names in a String array and the percentages in a double array (5 students, same index for the same student). Accept a roll number and, using linear search, display the student's name and percentage, or "Invalid Roll Number". **[7]**

Q10. Write a program to accept the elements of a 4 x 4 integer matrix and a number to be searched. Using linear search, display all the positions in the form (row, column) where the number is found, along with its total number of occurrences. **[7]**

Section B: Binary Search

Q11. Write a program to accept 10 integers in ascending order into an array and then accept a number to be searched. Using binary search, display its position (index + 1) or the message "Not Found". **[5]**

Q12. Write a program to store the names of 10 students in a String array in alphabetical order. Accept a name from the user and search for it using binary search (use compareToIgnoreCase()). Display the position if found. **[6]**

Q13. Write a program to accept the marks of 10 students in descending order. Accept a mark to be searched and use binary search (modified for descending order) to display its position, or "Not Found". **[6]**

Q14. Write a program that accepts 10 integers into an array. First check, using a loop, whether the array is in ascending order. If it is not sorted, display "Array not sorted" and stop. Otherwise, accept a number and search for it using a user-defined method int binSearch(int a[], int key). **[7]**

Q15. Write a program to accept 10 integers in ascending order and a number to be searched. Using binary search, display the position of the number (or "Not Found") and also the number of passes (comparisons with the middle element) the search took. **[6]**

Q16. Write a program to accept 8 letters in alphabetical order into a character array. Accept a letter from the user (convert it to uppercase) and search for it using binary search. Display whether it is present, and its position if found. **[6]**

Q17. Write a program to accept the prices (double) of 10 items in ascending order. Accept a price and search for it using binary search. If it is not found, display the position (index + 1) at which it should be inserted to keep the array sorted. **[7]**

Q18. A school stores 8 roll numbers in ascending order in an int array and the names of the corresponding students in a String array. Accept a roll number and, using binary search, display the name of the student, or "Invalid Roll Number". **[7]**

Q19. Write a program to accept 10 integers in ascending order (duplicates are allowed). Using a modified binary search, find and display the index of the first occurrence of a given number. (Hint: when the number is found at mid, record it and continue searching in the left half.) **[7]**

Q20. Write a program to store 8 English words in alphabetical order in a String array and their meanings in another String array (same index). Accept a word from the user and, using binary search, display its meaning, or "Word not found in the dictionary". **[8]**

Section C: Selection Sort

Q21. Write a program to accept 10 integers in an array and sort them in ascending order using selection sort. Display the sorted array. **[5]**

Q22. Write a program to accept the names of 8 students in a String array and arrange them in alphabetical order using selection sort (use compareTo()). Display the sorted list. **[5]**

Q23. Write a program to accept 10 integers and sort them in descending order using selection sort. Display the sorted array, and then display the largest and the second largest elements. **[6]**

Q24. Write a program to store the names of 6 students in a String array and their marks (out of 100) in a double array. Using selection sort, arrange the records in descending order of marks (the names must move along with their marks). Display the rank list in the form: Rank, Name, Marks. **[8]**

```
Sample Output:
Rank   Name     Marks
1      Meera    96.5
2      Karan    91.0
...
```

Q25. Write a program to accept a word, convert it to uppercase and store its letters in a character array. Arrange the letters in alphabetical order using selection sort and display the result. (Example: COMPUTER becomes CEMOPRTU) **[6]**

Q26. Write a program to accept 6 words in a String array. Using selection sort, arrange them in ascending order of their length. If two words have the same length, they should be arranged alphabetically. Display the sorted words along with their lengths. **[7]**

Q27. Write a program to accept 8 integers in an array. Using selection sort, sort them in ascending order and display the contents of the array after every pass. Use a user-defined method void selSort(int a[]) for sorting. **[7]**

Q28. Write a program to accept the monthly salaries (double) of 9 employees. Sort them in ascending order using selection sort and display the sorted salaries. Hence find and display the median salary (the middle value of the sorted array). **[7]**

Q29. Write a program to accept 10 integers and sort them in ascending order using selection sort. Along with the sorted array, display the total number of comparisons and the total number of exchanges (swaps) made during sorting. **[7]**

Q30. Write a program to accept 8 positive integers. Using selection sort, arrange them in ascending order of the sum of their digits. Display each number along with its digit sum. (Example: 91 has digit sum 10 and 23 has digit sum 5, so 23 comes before 91.) **[8]**

Section D: Bubble Sort

Q31. Write a program to accept 10 integers in an array and sort them in ascending order using bubble sort. Display the sorted array. **[5]**

Q32. Write a program to accept the names of 8 countries in a String array and arrange them in descending (reverse alphabetical) order using bubble sort. Display the sorted list. **[5]**

Q33. Write a program to accept 10 integers and sort them in ascending order using an improved bubble sort that uses a boolean flag to stop as soon as a pass makes no exchange. Display the sorted array and the number of passes actually taken. **[6]**

Q34. Write a program to accept a sentence. Store only the alphabets of the sentence (ignore spaces, digits and symbols, and convert to lowercase) in a character array. Arrange the characters in alphabetical order using bubble sort and display them. **[7]**

Q35. Write a program to store the names of 6 persons in a String array and their ages in an int array. Using bubble sort, arrange the records in ascending order of age (the names must move along with the ages) and display them in tabular form. **[8]**

Q36. Write a program to accept 6 words in a String array. Using bubble sort, arrange the words in descending order of the number of vowels they contain. Display each word along with its vowel count. **[7]**

Q37. Write a program to accept 10 integers (duplicates allowed). Sort them in ascending order using bubble sort and then display the array with all the duplicate elements removed (each distinct value shown once). **[7]**

```
Sample:
Input : 5 2 8 2 5 9 1 8 8 3
Output: 1 2 3 5 8 9
```

Q38. Write a program to store the roll numbers of 8 students in an int array and their marks (out of 100) in another int array. Using bubble sort, arrange the records in descending order of marks and display roll number, marks and grade. Grades: A for 90 and above, B for 75 to 89, C for 50 to 74, and D below 50. **[8]**

Q39. Write a program to accept 8 integers in an array. Define two user-defined methods: boolean isSorted(int a[]) that checks whether the array is in ascending order, and void bubble(int a[]) that sorts it using bubble sort. In main(), call bubble() only if the array is not already sorted, and display a suitable message and the final array. **[7]**

Q40. Write a program to accept the elements of a 4 x 4 integer matrix. Using bubble sort, arrange the elements of every row in ascending order and display the resulting matrix. **[8]**

Section E: Mixed Questions (Sorting followed by Searching)

Q41. Write a program to accept 10 integers in an unsorted array. Sort them in ascending order using selection sort, then accept a number and search for it using binary search. Display the sorted array and the position of the number in the sorted array, or "Not Found". **[8]**

Q42. Write a program to accept the names of 8 students in a String array. Arrange the names in alphabetical order using bubble sort. Then accept a name and search for it using binary search (ignore case). Display the sorted list and the position of the name. **[8]**

Q43. The marks of 10 students are entered in an array. Write a program to arrange the marks in descending order using bubble sort and then search a given mark using binary search (modified for descending order). If found, display the rank of the student (position in the sorted array). **[8]**

Q44. Write a program to store the names of 8 cities in a String array. Sort them alphabetically using selection sort, display them, and then use binary search to find whether a city entered by the user is present. Display "City found at position __" or "City not found". **[8]**

Q45. Write a menu-driven program to accept 10 integers in an array and perform the following as per the user's choice: (1) Sort using bubble sort, (2) Sort using selection sort, (3) Search using linear search, (4) Search using binary search. For choice 4, if the array has not been sorted by choice 1 or 2, display a message asking the user to sort it first. Display suitable messages for an invalid choice. **[10]**

Q46. Write a program to store the names of 6 students in a String array and their marks in an int array. Using selection sort, arrange the records in ascending order of marks (names must move along with marks). Then accept a student's name, search for it using linear search and display the student's marks and rank (the highest marks get rank 1). **[9]**

Q47. Write a program to accept a sentence (words separated by single spaces and ending with a full stop). Using charAt() and substring(), extract the words and store them in a String array. Sort the words alphabetically using bubble sort and display them. Then accept a word and search for it using binary search. **[9]**

Q48. Write a program to accept a word, convert it to uppercase and store its letters in a character array. Arrange the characters in alphabetical order using selection sort, then accept a character and check whether it is present using binary search. Display the sorted characters and the result. **[8]**

Q49. Write a program to accept the roll numbers (unsorted) of 8 students in an int array and their names in a String array. Using bubble sort, arrange the records in ascending order of roll number (names must move along). Then accept a roll number and, using binary search, display the name of the student or "Invalid Roll Number". **[9]**

Q50. Write a program to accept 10 integers in an array and a number x. Sort the array using selection sort and use binary search to check whether x is present. If x is absent, display the nearest smaller and the nearest larger elements present in the array (display "None" if either does not exist). **[9]**

**** End of Question Bank ****