

## Question - 1

```
1: #include <stdio.h>
2:
3: int main() {
4:     int arr[10], size = 0, choice;
5:     int i, pos, value, found;
6:
7:     while (1) {
8:         printf("MENU\n");
9:         printf("1. CREATE\n");
10:        printf("2. DISPLAY\n");
11:        printf("3. INSERT\n");
12:        printf("4. DELETE\n");
13:        printf("5. LINEAR SEARCH\n");
14:        printf("6. EXIT\n");
15:        printf("Enter your choice: ");
16:        scanf("%d", &choice);
17:
18:        if (choice == 1) {
19:            printf("Enter number of elements: ");
20:            scanf("%d", &size);
21:            printf("Enter %d elements:\n", size);
22:            for (i = 0; i < size; i++) {
23:                scanf("%d", &arr[i]);
24:            }
25:        }
26:        else if (choice == 2) {
27:            if (size == 0) {
28:                printf("Array is empty.\n");
29:            } else {
30:                printf("Array elements are:\n");
31:                for (i = 0; i < size; i++) {
32:                    printf("%d ", arr[i]);
33:                }
34:                printf("\n");
35:            }
36:        }
37:        else if (choice == 3) {
38:            if (size >= 10) {
39:                printf("Array is full. Cannot insert.\n");
40:                continue;
41:            }
42:            printf("Enter position to insert (0 to %d): ", size);
43:            scanf("%d", &pos);
44:            if (pos < 0 || pos > size) {
45:                printf("Invalid position.\n");
46:            } else {
47:                printf("Enter value to insert: ");
48:                scanf("%d", &value);
49:                for (i = size; i > pos; i--) {
50:                    arr[i] = arr[i - 1];
51:                }
52:                arr[pos] = value;
53:                size++;
54:                printf("Value inserted.\n");
55:            }
56:        }
57:    }
58: }
```

```

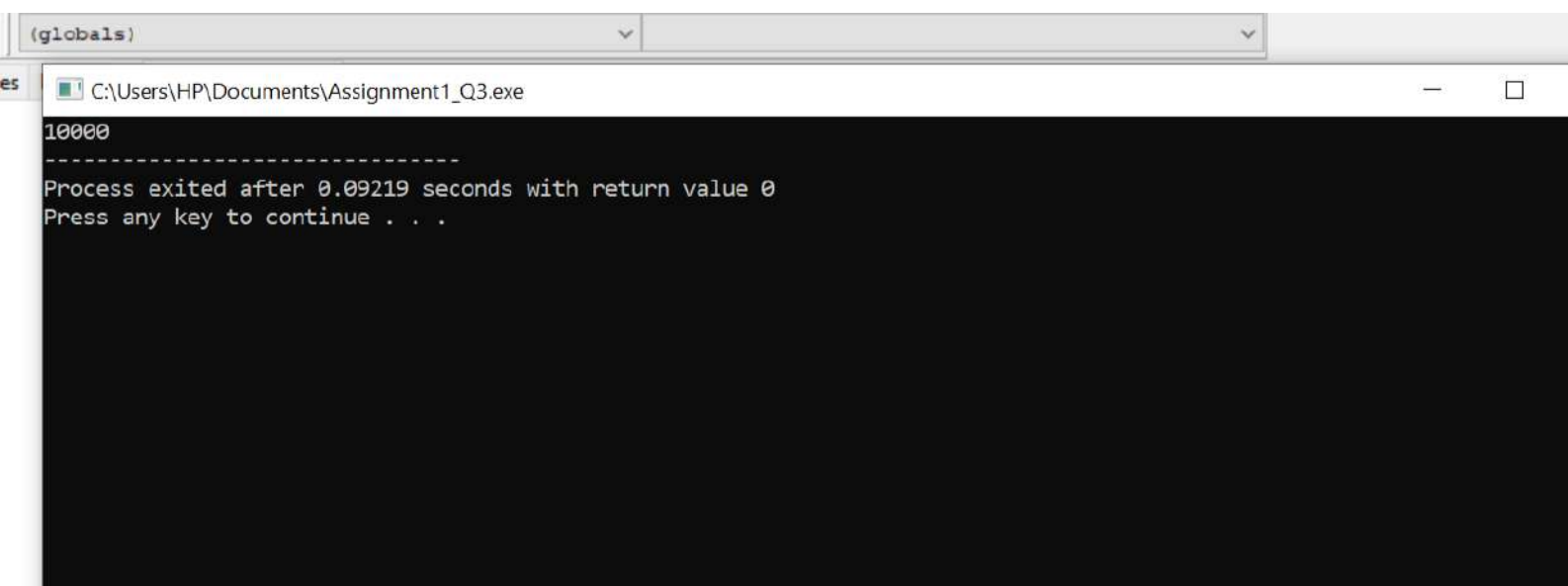
56:     }
57:     else if (choice == 4) {
58:         if (size == 0) {
59:             printf("Array is empty. Nothing to delete.\n");
60:             continue;
61:         }
62:         printf("Enter position to delete (0 to %d): ", size - 1);
63:         scanf("%d", &pos);
64:         if (pos < 0 || pos >= size) {
65:             printf("Invalid position.\n");
66:         } else {
67:             for (i = pos; i < size - 1; i++) {
68:                 arr[i] = arr[i + 1];
69:             }
70:             size--;
71:             printf("Element deleted.\n");
72:         }
73:     }
74:     else if (choice == 5) {
75:         if (size == 0) {
76:             printf("Array is empty.\n");
77:             continue;
78:         }
79:         printf("Enter value to search: ");
80:         scanf("%d", &value);
81:         found = 0;
82:         for (i = 0; i < size; i++) {
83:             if (arr[i] == value) {
84:                 printf("Value found at position %d\n", i);
85:                 found = 1;
86:                 break;
87:             }
88:         }
89:         if (!found) {
90:             printf("Value not found in array.\n");
91:         }
92:     }
93:     else if (choice == 6) {
94:         printf("Exiting program.\n");
95:         break;
96:     }
97:     else {
98:         printf("Invalid choice. Please try again.\n");
99:     }
100: }
101:
102: return 0;
103: }
104:

```

## Question - 2

```
1: #include <stdio.h>
2:
3: int main() {
4:     int arr[10], n;
5:
6:     printf("Enter number of elements: ");
7:     scanf("%d", &n);
8:
9:     printf("Enter %d elements:\n", n);
10:    for (int i = 0; i < n; i++) {
11:        scanf("%d", &arr[i]);
12:    }
13:
14:    for (int i = 0; i < n; i++) {
15:        for (int j = i + 1; j < n; ) {
16:            if (arr[i] == arr[j]) {
17:                for (int k = j; k < n - 1; k++) {
18:                    arr[k] = arr[k + 1];
19:                }
20:                n--;
21:            }
22:            else {
23:                j++;
24:            }
25:        }
26:    }
27:    printf("Array after removing duplicates: ");
28:    for (int i = 0; i < n; i++) {
29:        printf("%d ", arr[i]);
30:    }
31:    printf("\n");
32:
33:    return 0;
34: }
35:
```

### Question - 3



The screenshot shows a Windows command prompt window with a title bar that includes the file path "C:\Users\HP\Documents\Assignment1\_Q3.exe". The window has a menu bar with "File", "Edit", and "Format" options. The command prompt shows the following text:

```
(globals)
es
C:\Users\HP\Documents\Assignment1_Q3.exe
10000
-----
Process exited after 0.09219 seconds with return value 0
Press any key to continue . . .
```

#### Question - 4

```
1: #include <stdio.h>
2:
3: int main() {
4:     int choice;
5:
6:     printf("MENU\n");
7:     printf("1. Reverse an Array\n");
8:     printf("2. Matrix Multiplication\n");
9:     printf("3. Transpose of a Matrix\n");
10:    printf("Enter your choice: ");
11:    scanf("%d", &choice);
12:
13:    if (choice == 1) {
14:        int arr[100], n, i, temp;
15:        printf("Enter size of array: ");
16:        scanf("%d", &n);
17:        printf("Enter %d elements:\n", n);
18:        for (i = 0; i < n; i++) {
19:            scanf("%d", &arr[i]);
20:        }
21:
22:        for (i = 0; i < n / 2; i++) {
23:            temp = arr[i];
24:            arr[i] = arr[n - 1 - i];
25:            arr[n - 1 - i] = temp;
26:        }
27:
28:        printf("Reversed array:\n");
29:        for (i = 0; i < n; i++) {
30:            printf("%d ", arr[i]);
31:        }
32:        printf("\n");
33:    }
34:
35:    else if (choice == 2) {
36:        int a[10][10], b[10][10], result[10][10];
37:        int r1, c1, r2, c2;
38:        int i, j, k;
39:
40:        printf("Enter rows and columns for Matrix A: ");
41:        scanf("%d%d", &r1, &c1);
42:        printf("Enter rows and columns for Matrix B: ");
43:        scanf("%d%d", &r2, &c2);
44:
45:        if (c1 != r2) {
46:            printf("Matrix multiplication not possible. Columns of A must equal rows of B.\n");
47:        } else {
48:            printf("Enter elements of Matrix A:\n");
49:            for (i = 0; i < r1; i++) {
50:                for (j = 0; j < c1; j++) {
51:                    scanf("%d", &a[i][j]);
52:                }
53:            }
54:            printf("Enter elements of Matrix B:\n");
55:            for (i = 0; i < r2; i++) {
```

```

56:         for (j = 0; j < c2; j++) {
57:             scanf("%d", &b[i][j]);
58:         }
59:     }
60:     for (i = 0; i < r1; i++) {
61:         for (j = 0; j < c2; j++) {
62:             result[i][j] = 0;
63:             for (k = 0; k < c1; k++) {
64:                 result[i][j] += a[i][k] * b[k][j];
65:             }
66:         }
67:     }
68:     printf("Resultant Matrix:\n");
69:     for (i = 0; i < r1; i++) {
70:         for (j = 0; j < c2; j++) {
71:             printf("%d ", result[i][j]);
72:         }
73:         printf("\n");
74:     }
75: }
76: }
77: else if (choice == 3) {
78:     int mat[10][10], trans[10][10];
79:     int rows, cols;
80:     int i, j;
81:     printf("Enter rows and columns of matrix: ");
82:     scanf("%d%d", &rows, &cols);
83:     printf("Enter elements of matrix:\n");
84:     for (i = 0; i < rows; i++) {
85:         for (j = 0; j < cols; j++) {
86:             scanf("%d", &mat[i][j]);
87:         }
88:     }
89:     for (i = 0; i < rows; i++) {
90:         for (j = 0; j < cols; j++) {
91:             trans[j][i] = mat[i][j];
92:         }
93:     }
94:     printf("Transpose of matrix:\n");
95:     for (i = 0; i < cols; i++) {
96:         for (j = 0; j < rows; j++) {
97:             printf("%d ", trans[i][j]);
98:         }
99:         printf("\n");
100:     }
101: }
102: else {
103:     printf("Invalid choice.\n");
104: }
105: return 0;
106: }
107:

```

## Question - 5

```
1: #include <stdio.h>
2:
3: int main() {
4:     int rows, cols;
5:     int i, j;
6:     int arr[5][5];
7:
8:     printf("Enter number of rows: ");
9:     scanf("%d", &rows);
10:
11:     printf("Enter number of columns: ");
12:     scanf("%d", &cols);
13:
14:     printf("Enter elements of the matrix:\n");
15:     for (i = 0; i < rows; i++) {
16:         for (j = 0; j < cols; j++) {
17:             printf("Element [%d][%d]: ", i, j);
18:             scanf("%d", &arr[i][j]);
19:         }
20:     }
21:
22:     printf("\nSum of each row:\n");
23:     for (i = 0; i < rows; i++) {
24:         int rowSum = 0;
25:         for (j = 0; j < cols; j++) {
26:             rowSum += arr[i][j];
27:         }
28:         printf("Row %d sum = %d\n", i + 1, rowSum);
29:     }
30:
31:     printf("\nSum of each column:\n");
32:     for (j = 0; j < cols; j++) {
33:         int colSum = 0;
34:         for (i = 0; i < rows; i++) {
35:             colSum += arr[i][j];
36:         }
37:         printf("Column %d sum = %d\n", j + 1, colSum);
38:     }
39:
40:     return 0;
41: }
42:
```