1. WAP to insert an element at the beginning of an array. 2. WAP to insert an element at the end of an array. 3. WAP to insert an element at the nth position of an array. 4. WAP to delete an element from the beginning of an array. 5. WAP to delete the last element of an array. 6. WAP to delete the nth element of an array. 7. WAP to multiply 2 matrices. 8. WAP to find the number of characters in a string. 9. WAP to reverse a string. 10. WAP to convert lower case characters of a string to upper case. give code for those and i dont want psudo code like that i need pseudo code it like code form and explained , not in words like you are doing

2/2

ChatGPT said:

ChatGPT

Memory updated

Here are the solutions in C, along with pseudocode in a code-like format with explanations.

---

1. Insert an Element at the Beginning of an Array

Pseudocode:

c

Copy code
```
// Initialize array with size and elements, specify new element to
insert
int arr[] = {2, 3, 4, 5};
int n = 4;  // current size
int new_element = 1;

// Shift all elements to the right by one position to make space at
the beginning
for (int i = n; i > 0; i--) {
    arr[i] = arr[i - 1];
}

// Insert new element at index 0 and increment size
arr[0] = new_element;
n++;
```

Code:

c

```c
#include <stdio.h>

int main() {
    int arr[100] = {2, 3, 4, 5};
    int n = 4, new_element = 1;

    for (int i = n; i > 0; i--) {
        arr[i] = arr[i - 1];
    }
    arr[0] = new_element;
    n++;

    printf("Array after insertion: ");
    for (int i = 0; i < n; i++) {
        printf("%d ", arr[i]);
    }
    return 0;
}
```

---

2. Insert an Element at the End of an Array

Pseudocode:

c

```c
// Initialize array and specify new element to insert
int arr[] = {2, 3, 4, 5};
int n = 4;   // current size
int new_element = 6;

// Add new element at index `n` and increment size
arr[n] = new_element;
n++;
```

Code:

c

Copy code

```
#include <stdio.h>

int main() {
    int arr[100] = {2, 3, 4, 5};
    int n = 4, new_element = 6;

    arr[n] = new_element;
    n++;

    printf("Array after insertion: ");
    for (int i = 0; i < n; i++) {
        printf("%d ", arr[i]);
    }
    return 0;
}
```

---

3. Insert an Element at the nth Position of an Array

Pseudocode:

c
Copy code

```
// Initialize array, specify position `pos` and new element to insert
int arr[] = {2, 3, 4, 5};
int n = 4;   // current size
int pos = 2, new_element = 99;

// Shift elements to the right starting from last element up to
position `pos`
for (int i = n; i > pos; i--) {
    arr[i] = arr[i - 1];
}

// Insert new element at the specified position and increment size
arr[pos] = new_element;
n++;
```

Code:

c
Copy code

```c
#include <stdio.h>

int main() {
    int arr[100] = {2, 3, 4, 5};
    int n = 4, pos = 2, new_element = 99;

    for (int i = n; i > pos; i--) {
        arr[i] = arr[i - 1];
    }
    arr[pos] = new_element;
    n++;

    printf("Array after insertion: ");
    for (int i = 0; i < n; i++) {
        printf("%d ", arr[i]);
    }
    return 0;
}
```

---

4. Delete an Element from the Beginning of an Array

Pseudocode:

c
Copy code

```c
// Initialize array with elements
int arr[] = {2, 3, 4, 5};
int n = 4;  // current size

// Shift elements to the left starting from index 1
for (int i = 1; i < n; i++) {
    arr[i - 1] = arr[i];
}
```

```c
// Decrease size
n--;
```

Code:

c
Copy code
```c
#include <stdio.h>

int main() {
    int arr[100] = {2, 3, 4, 5};
    int n = 4;

    for (int i = 1; i < n; i++) {
        arr[i - 1] = arr[i];
    }
    n--;

    printf("Array after deletion: ");
    for (int i = 0; i < n; i++) {
        printf("%d ", arr[i]);
    }
    return 0;
}
```

---

5. Delete the Last Element of an Array

Pseudocode:

c
Copy code
```c
// Initialize array with size `n`
int arr[] = {2, 3, 4, 5};
int n = 4;

// Just decrement the size to "remove" the last element
n--;
```

Code:

c
Copy code

```c
#include <stdio.h>

int main() {
    int arr[100] = {2, 3, 4, 5};
    int n = 4;

    n--;

    printf("Array after deletion: ");
    for (int i = 0; i < n; i++) {
        printf("%d ", arr[i]);
    }
    return 0;
}
```

---

6. Delete the nth Element of an Array

Pseudocode:

c
Copy code

```c
// Initialize array and specify position `pos` of element to delete
int arr[] = {2, 3, 4, 5};
int n = 4;   // current size
int pos = 2;

// Shift elements to the left starting from `pos + 1` to overwrite the element at `pos`
for (int i = pos; i < n - 1; i++) {
    arr[i] = arr[i + 1];
}

// Decrease size
n--;
```

Code:

c
Copy code

```c
#include <stdio.h>

int main() {
    int arr[100] = {2, 3, 4, 5};
    int n = 4, pos = 2;

    for (int i = pos; i < n - 1; i++) {
        arr[i] = arr[i + 1];
    }
    n--;

    printf("Array after deletion: ");
    for (int i = 0; i < n; i++) {
        printf("%d ", arr[i]);
    }
    return 0;
}
```

---

7. Multiply Two Matrices

Pseudocode:

c
Copy code

```c
// Initialize matrices `A` and `B` with dimensions and result matrix
`C`
int A[2][2] = {{1, 2}, {3, 4}};
int B[2][2] = {{5, 6}, {7, 8}};
int C[2][2] = {0};

// Multiply A and B, store the result in C
for (int i = 0; i < 2; i++) {
    for (int j = 0; j < 2; j++) {
        for (int k = 0; k < 2; k++) {
```

```c
            C[i][j] += A[i][k] * B[k][j];
        }
    }
}
```

Code:

c
Copy code
```c
#include <stdio.h>

int main() {
    int A[2][2] = {{1, 2}, {3, 4}};
    int B[2][2] = {{5, 6}, {7, 8}};
    int C[2][2] = {0};

    for (int i = 0; i < 2; i++) {
        for (int j = 0; j < 2; j++) {
            for (int k = 0; k < 2; k++) {
                C[i][j] += A[i][k] * B[k][j];
            }
        }
    }

    printf("Result matrix:\n");
    for (int i = 0; i < 2; i++) {
        for (int j = 0; j < 2; j++) {
            printf("%d ", C[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```

8. Find the Number of Characters in a String

Pseudocode:

c
Copy code

```c
// Initialize a string
char str[] = "Hello World";
int count = 0;

// Loop through each character until null terminator is found
while (str[count] != '\0') {
    count++;
}
```

Code:

c
Copy code

```c
#include <stdio.h>

int main() {
    char str[] = "Hello World";
    int count = 0;

    while (str[count] != '\0') {
        count++;
    }

    printf("Number of characters: %d\n", count);
    return 0;
}
```

---

9. Reverse a String (Continued)

Pseudocode:

c
Copy code

```c
// Initialize string to be reversed
```

```c
char str[] = "Hello";
int n = strlen(str);

// Swap characters from both ends until reaching the middle of the
string
for (int i = 0; i < n / 2; i++) {
    char temp = str[i];              // temporarily hold the character
    str[i] = str[n - i - 1];         // swap current character with the
mirrored character
    str[n - i - 1] = temp;           // place held character in the
mirrored position
}
```

Code:

c
Copy code
```c
#include <stdio.h>
#include <string.h>

int main() {
    char str[] = "Hello";
    int n = strlen(str);

    for (int i = 0; i < n / 2; i++) {
        char temp = str[i];
        str[i] = str[n - i - 1];
        str[n - i - 1] = temp;
    }

    printf("Reversed string: %s\n", str);
    return 0;
}
```

10. Convert Lowercase Characters of a String to Uppercase

Pseudocode:

c
Copy code

```c
// Initialize the string
char str[] = "Hello World";

// Loop through each character in the string
for (int i = 0; str[i] != '\0'; i++) {
    // Check if the character is lowercase
    if (str[i] >= 'a' && str[i] <= 'z') {
        // Convert to uppercase by subtracting ASCII offset
        str[i] = str[i] - ('a' - 'A');
    }
}
```

Code:

c
Copy code

```c
#include <stdio.h>

int main() {
    char str[] = "Hello World";

    for (int i = 0; str[i] != '\0'; i++) {
        if (str[i] >= 'a' && str[i] <= 'z') {
            str[i] = str[i] - ('a' - 'A');
        }
    }

    printf("Uppercase string: %s\n", str);
    return 0;
}
```