

1. The output of the code snippet is undefined. It invokes undefined behavior due to multiple modifications and lack of sequence points in the expression.
2. Recursion is a technique where a function calls itself to solve a problem. Example:

```
#include <stdio.h>

int factorial(int n) {
    if (n == 0)
        return 1;
    else
        return n * factorial(n - 1);
}

int main() {
    int num = 5;
    int result = factorial(num);
    printf("Factorial of %d is %d", num, result);
    return 0;
}
```

3. The **volatile** keyword is used to indicate that a variable may be modified by external entities, preventing certain optimizations by the compiler.

4. A structure in C is a collection of different data types grouped together, while a union is a data type that can hold variables of different types, but only one member can be accessed at a time.
5. The **const** keyword in C is used to declare a variable as read-only, meaning its value cannot be modified. When used with pointer variables, it ensures that the pointer itself cannot be used to modify the value it points to.
6. Here's a code snippet to reverse a linked list in C:

```
#include <stdio.h>
#include <stdlib.h>
typedef struct Node {
    int data;
    struct Node* next;
} Node;
void reverseLinkedList(Node** head) {
    Node* prev = NULL;
    Node* current = *head;
    Node* next = NULL;
    while (current != NULL) {
        next = current->next;
        current->next = prev;
        prev = current;
```

```

    current = next;
}
*head = prev;
}
int main() {
    Node* head = NULL;
    // Linked list creation and insertion code goes here...
    reverseLinkedList(&head);
    // Print the reversed linked list...
    return 0;
}

```

7. Function pointers in C are variables that store the memory address of a function. They can be used to dynamically select and invoke functions at runtime.  
Example:

```
#include <stdio.h>
```

```

void add(int a, int b) {
    printf("Sum: %d", a + b);
}

```

```
void subtract(int a, int b) {  
    printf("Difference: %d", a - b);  
}
```

```
int main() {  
    void (*funcPtr)(int, int);  
    int choice = 1;  
    if (choice == 1)  
        funcPtr = add;  
    else  
        funcPtr = subtract;  
    funcPtr(5, 3); // Invokes the selected function  
    return 0;  
}
```

8. The output of the code snippet is undefined. It invokes undefined behavior by accessing memory beyond the array boundaries, leading to unpredictable results.