

1. Write a C program to calculate and display the area of a rectangle using the input values entered by the user.

Question: How do you calculate the area of a rectangle in C?

Answer: The area of a rectangle can be calculated using the formula: $\text{area} = \text{length} * \text{width}$. Here's a sample program:

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

int main() {
    float length, width, area;
    printf("Enter the length of the rectangle: ");
    scanf("%f", &length);
    printf("Enter the width of the rectangle: ");
    scanf("%f", &width);
    area = length * width;
    printf("The area of the rectangle is: %.2f\n", area);
    return 0;
}
```

2. Write a C program to sort an array of integers using the selection sort technique.

Question: How does the selection sort algorithm work?

Answer: The selection sort algorithm repeatedly finds the minimum element from the unsorted part and places it at the beginning. Here's a program to sort an array using selection sort:

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

void selectionSort(int arr[], int n) {
    int i, j, minIndex, temp;
    for (i = 0; i < n - 1; i++) {
        minIndex = i;
        for (j = i + 1; j < n; j++) {
            if (arr[j] < arr[minIndex]) {
                minIndex = j;
            }
        }
        temp = arr[minIndex];
        arr[minIndex] = arr[i];
        arr[i] = temp;
    }
}

int main() {
    int arr[5], i;
    printf("Enter 5 integers: ");
    for(i = 0; i < 5; i++)
```

```
        scanf("%d", &arr[i]);
    selectionSort(arr, 5);
    printf("Sorted array: ");
    for(i = 0; i < 5; i++)
        printf("%d ", arr[i]);
    return 0;
}
```

3. Write a C program to concatenate two strings entered by the user and display the resultant string.

Question: How do you concatenate two strings in C without using library functions?

Answer: You can concatenate two strings by appending each character of the second string to the end of the first string manually. Here's how:

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

void concatenate(char str1[], char str2[]) {
    int i = 0, j = 0;
    while (str1[i] != '\0') i++;
    while (str2[j] != '\0') {
        str1[i] = str2[j];
        i++;
        j++;
    }
    str1[i] = '\0';
}

int main() {
    char str1[100], str2[50];
    printf("Enter first string: ");
    gets(str1);
    printf("Enter second string: ");
    gets(str2);
    concatenate(str1, str2);
    printf("Concatenated string: %s\n", str1);
    return 0;
}
```

4. Write a C program to find the factorial of a number using recursion.

Question: How can you find the factorial of a number using recursion?

Answer: A factorial function calls itself with a decremented value until it reaches 1. Here's an example:

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

int factorial(int n) {
```

```

        if (n == 1)
            return 1;
        return n * factorial(n - 1);
    }

int main() {
    int num;
    printf("Enter a number: ");
    scanf("%d", &num);
    printf("Factorial of %d is %d\n", num, factorial(num));
    return 0;
}

```

5. Write a C program to swap two numbers using call by value and call by reference.

Question: What is the difference between call by value and call by reference?

Answer: In call by value, a copy of the actual parameter is passed, whereas in call by reference, the address of the actual parameter is passed, allowing modification. Here's an example:

```

c
Copy code
#include <stdio.h>

void swapByValue(int a, int b) {
    int temp = a;
    a = b;
    b = temp;
    printf("After swap by value: a = %d, b = %d\n", a, b);
}

void swapByReference(int *a, int *b) {
    int temp = *a;
    *a = *b;
    *b = temp;
}

int main() {
    int x = 10, y = 20;
    printf("Before swap: x = %d, y = %d\n", x, y);
    swapByValue(x, y);
    swapByReference(&x, &y);
    printf("After swap by reference: x = %d, y = %d\n", x, y);
    return 0;
}

```

6. Write a C program to create a structure named student with the fields roll no, name, and marks in three subjects.

Question: How do you use structures in C to store and display student information?

Answer: You can define a structure with the required fields and use it to store data for a student. Here's an example:

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

struct Student {
    int rollNo;
    char name[50];
    float marks[3];
};

int main() {
    struct Student s;
    printf("Enter roll number: ");
    scanf("%d", &s.rollNo);
    printf("Enter name: ");
    scanf("%s", s.name);
    for(int i = 0; i < 3; i++) {
        printf("Enter marks in subject %d: ", i + 1);
        scanf("%f", &s.marks[i]);
    }
    printf("Student Details:\nRoll No: %d\nName: %s\n", s.rollNo, s.name);
    for(int i = 0; i < 3; i++)
        printf("Marks in subject %d: %.2f\n", i + 1, s.marks[i]);
    return 0;
}
```

7. Write a C program to read data from a text file and display it on the screen.

Question: How do you read from a file in C?

Answer: You can use `fopen`, `fgets`, and `fclose` to read data from a text file. Here's an example:

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

int main() {
    FILE *file = fopen("data.txt", "r");
    char line[100];
    if (file == NULL) {
        printf("Error opening file\n");
        return 1;
    }
    while (fgets(line, sizeof(line), file)) {
        printf("%s", line);
    }
    fclose(file);
    return 0;
}
```

8. Write a C program to implement a singly linked list and display its elements.

Question: How do you create and display elements in a singly linked list?

Answer: You can define a node structure and use pointers to link nodes together. Here's a basic linked list program:

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

struct Node {
    int data;
    struct Node* next;
};

void display(struct Node* head) {
    struct Node* temp = head;
    while (temp != NULL) {
        printf("%d -> ", temp->data);
        temp = temp->next;
    }
    printf("NULL\n");
}

int main() {
    struct Node* head = malloc(sizeof(struct Node));
    struct Node* second = malloc(sizeof(struct Node));
    struct Node* third = malloc(sizeof(struct Node));

    head->data = 1;
    head->next = second;

    second->data = 2;
    second->next = third;

    third->data = 3;
    third->next = NULL;

    display(head);

    free(head);
    free(second);
    free(third);

    return 0;
}
```

9. Write a C program to open a binary file, write data to it, and read data from it.

Question: How do you write to and read from a binary file in C?

Answer: You can use `fwrite` and `fread` functions for binary file operations. Here's an example:

C

Copy code

```
#include <stdio.h>

int main() {
    FILE *file = fopen("data.bin", "wb");
    int num = 12345;
    fwrite(&num, sizeof(int), 1, file);
    fclose(file);

    file = fopen("data.bin", "rb");
    fread(&num, sizeof(int), 1, file);
    printf("Read from binary file: %d\n", num);
    fclose(file);

    return 0;
}
```

10. Write a C program to implement a stack using an array and perform push, pop, and display.

Question: How can you implement a stack using an array in C?

Answer: You can use an array to represent the stack and functions to perform push, pop, and display operations. Here's an example:

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

#define MAX 5
int stack[MAX], top = -1;

void push(int val) {
    if (top == MAX - 1) printf("Stack overflow\n");
    else stack[++top] = val;
}

int pop() {
    if (top == -1) printf("Stack underflow\n");
    else return stack[top--];
    return -1;
}

void display() {
    if (top == -1) printf("Stack is empty\n");
    else {
        for (int i = top; i >= 0; i--)
            printf("%d ", stack[i]);
        printf("\n");
    }
}

int main() {
    push(10);
    push(20);
    push(30);
    display();
    printf("Popped: %d\n", pop());
    display();
    return 0;
}
```