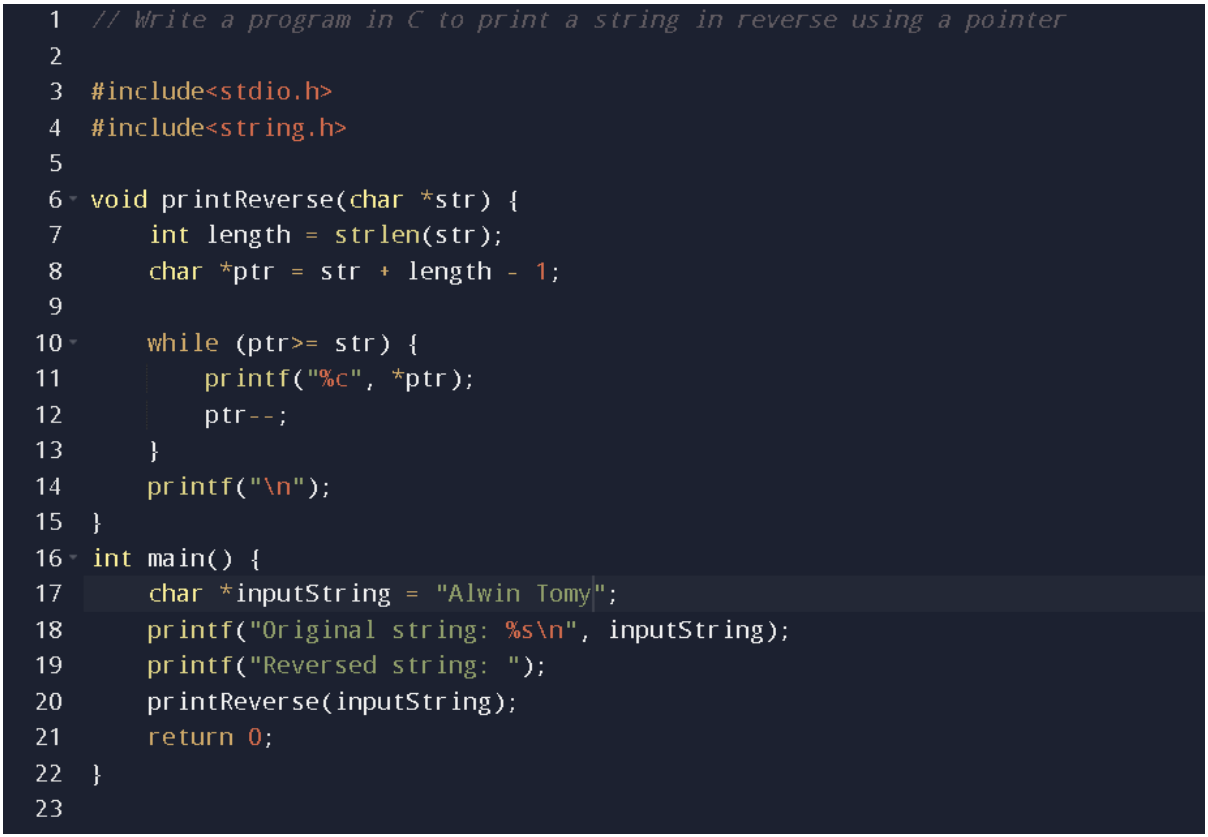
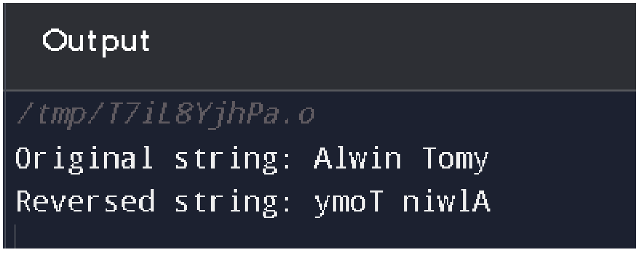
Alwin Tomy (2347207)

1. Write a program in C to print a string in reverse using a pointer.

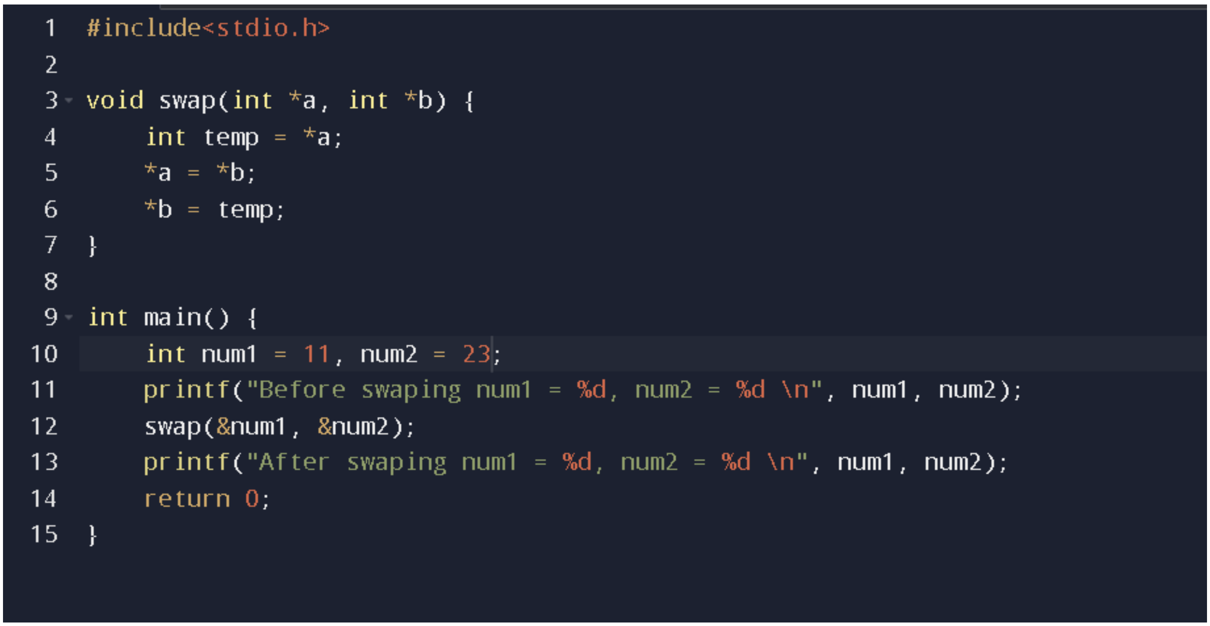


```c
// Write a program in C to print a string in reverse using a pointer

#include<stdio.h>
#include<string.h>

void printReverse(char *str) {
    int length = strlen(str);
    char *ptr = str + length - 1;

    while (ptr>= str) {
        printf("%c", *ptr);
        ptr--;
    }
    printf("\n");
}
int main() {
    char *inputString = "Alwin Tomy";
    printf("Original string: %s\n", inputString);
    printf("Reversed string: ");
    printReverse(inputString);
    return 0;
}

```

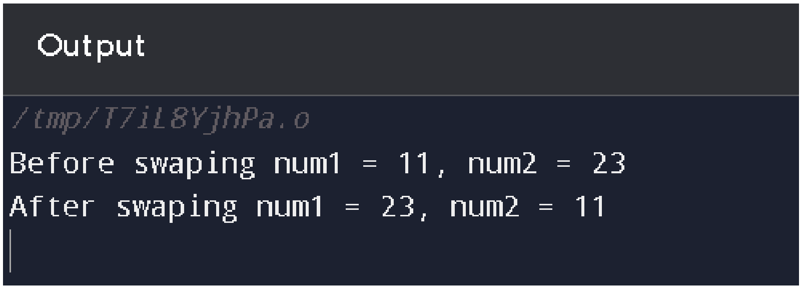


```
/tmp/T7iL8YjhPa.o
Original string: Alwin Tomy
Reversed string: ymoT niwlA
```

2. Write a program in C to swap elements using call by reference.

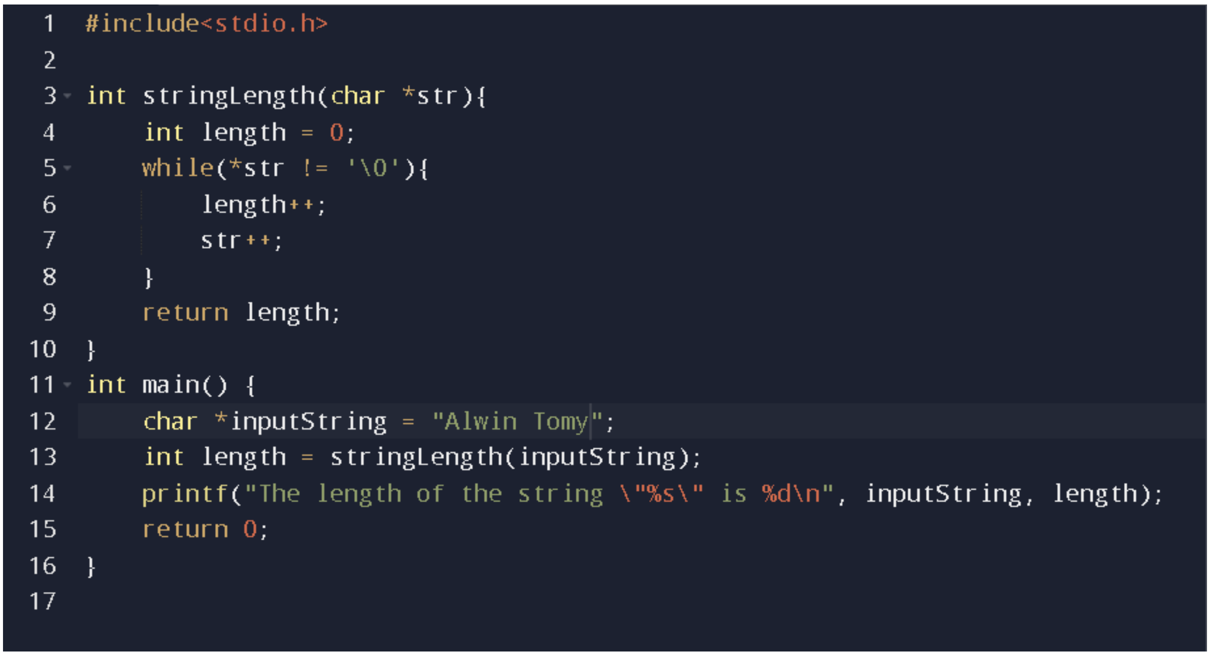


```c
#include<stdio.h>

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

int main() {
    int num1 = 11, num2 = 23;
    printf("Before swaping num1 = %d, num2 = %d \n", num1, num2);
    swap(&num1, &num2);
    printf("After swaping num1 = %d, num2 = %d \n", num1, num2);
    return 0;
}
```

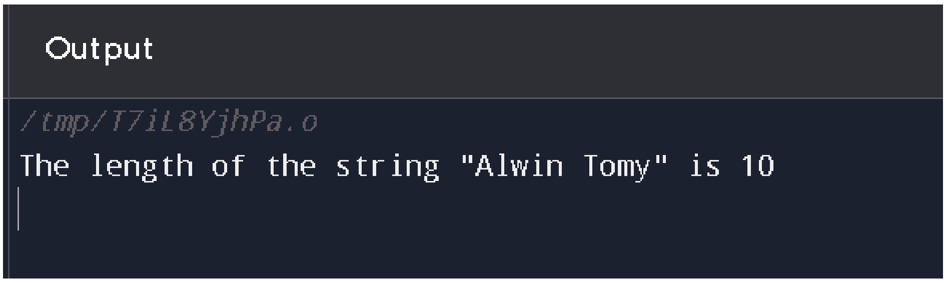


```
/tmp/T7iL8YjhPa.o
Before swaping num1 = 11, num2 = 23
After swaping num1 = 23, num2 = 11
```

3. Write a program in C to calculate the length of a string using a pointer.

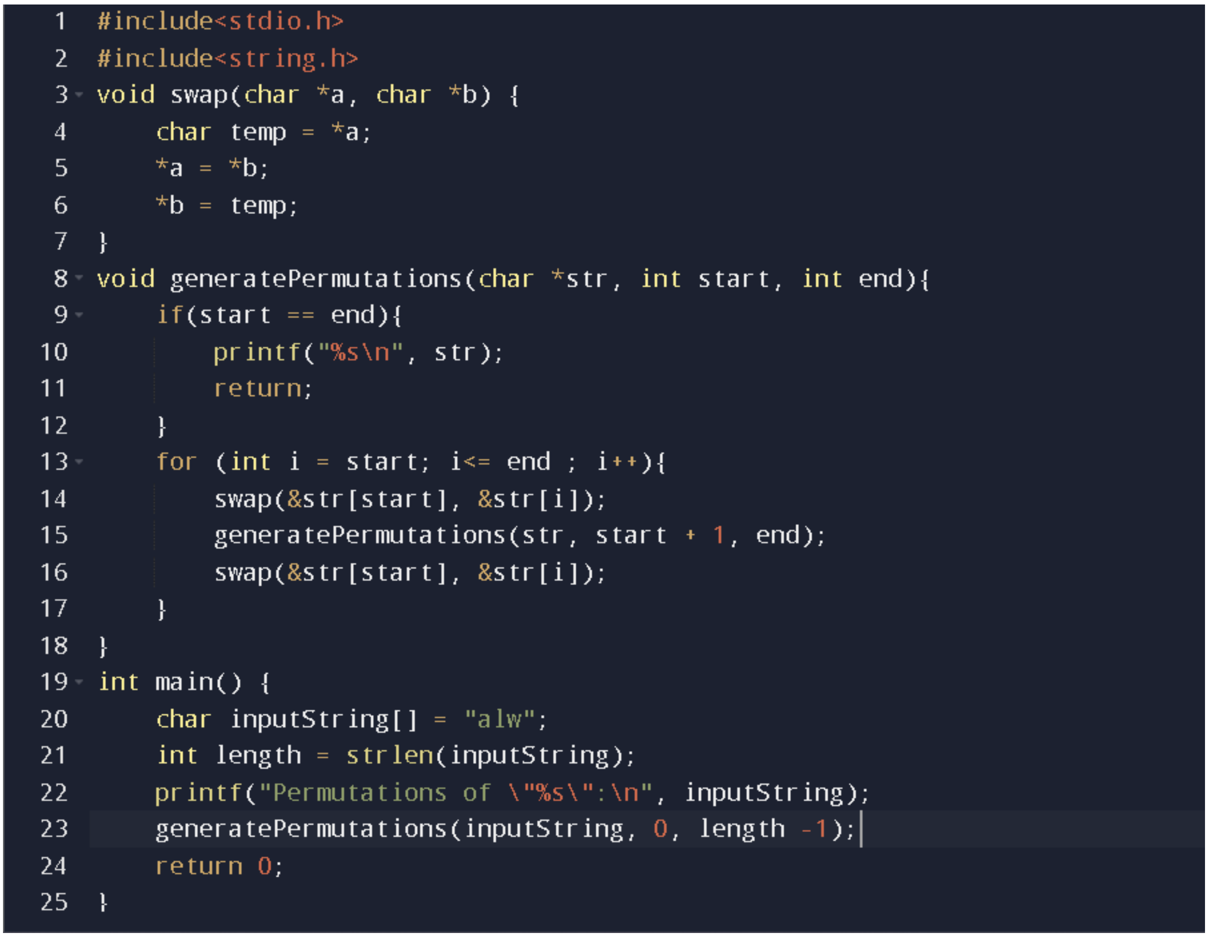


```c
#include<stdio.h>

int stringLength(char *str){
    int length = 0;
    while(*str != '\0'){
        length++;
        str++;
    }
    return length;
}
int main() {
    char *inputString = "Alwin Tomy";
    int length = stringLength(inputString);
    printf("The length of the string \"%s\" is %d\n", inputString, length);
    return 0;
}
```

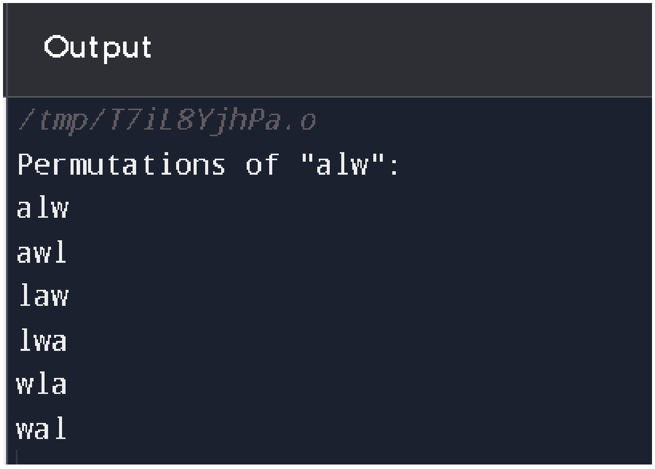


```
/tmp/T7iL8YjhPa.o
The length of the string "Alwin Tomy" is 10
```

4. Write a program in C to print all permutations of a given string using pointers.

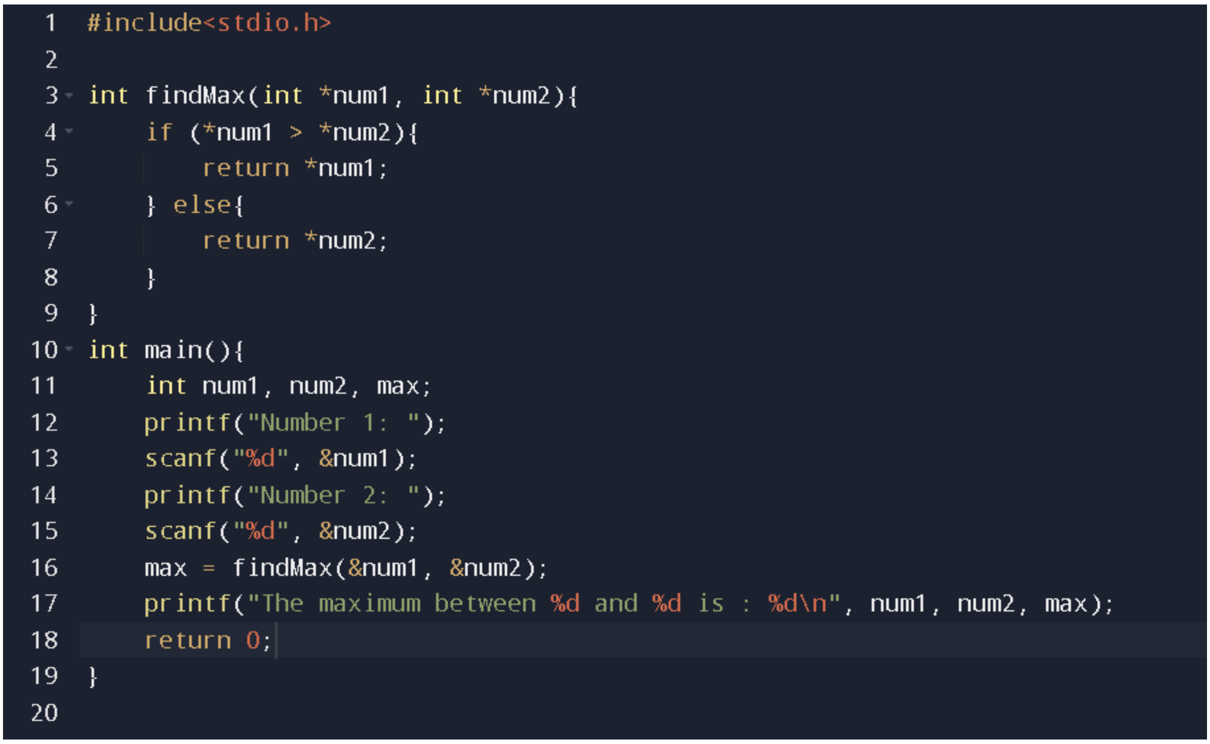


```c
#include<stdio.h>
#include<string.h>
void swap(char *a, char *b) {
    char temp = *a;
    *a = *b;
    *b = temp;
}
void generatePermutations(char *str, int start, int end){
    if(start == end){
        printf("%s\n", str);
        return;
    }
    for (int i = start; i<= end ; i++){
        swap(&str[start], &str[i]);
        generatePermutations(str, start + 1, end);
        swap(&str[start], &str[i]);
    }
}
int main() {
    char inputString[] = "alw";
    int length = strlen(inputString);
    printf("Permutations of \"%s\":\n", inputString);
    generatePermutations(inputString, 0, length -1);
    return 0;
}
```

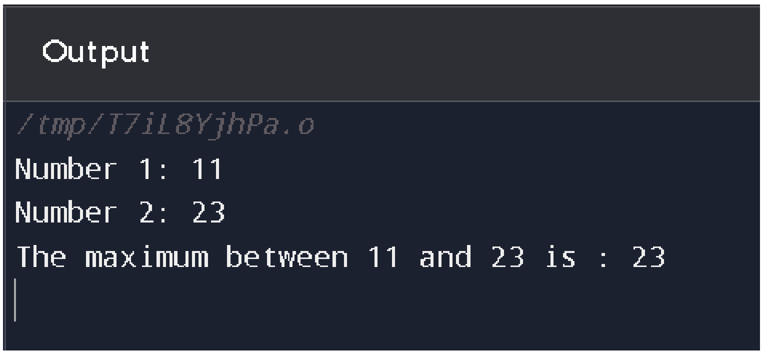


```
/tmp/T7iL8YjhPa.o
Permutations of "alw":
alw
awl
law
lwa
wla
wal
```

5. Write a program in C to find the maximum number between two numbers using a pointer.



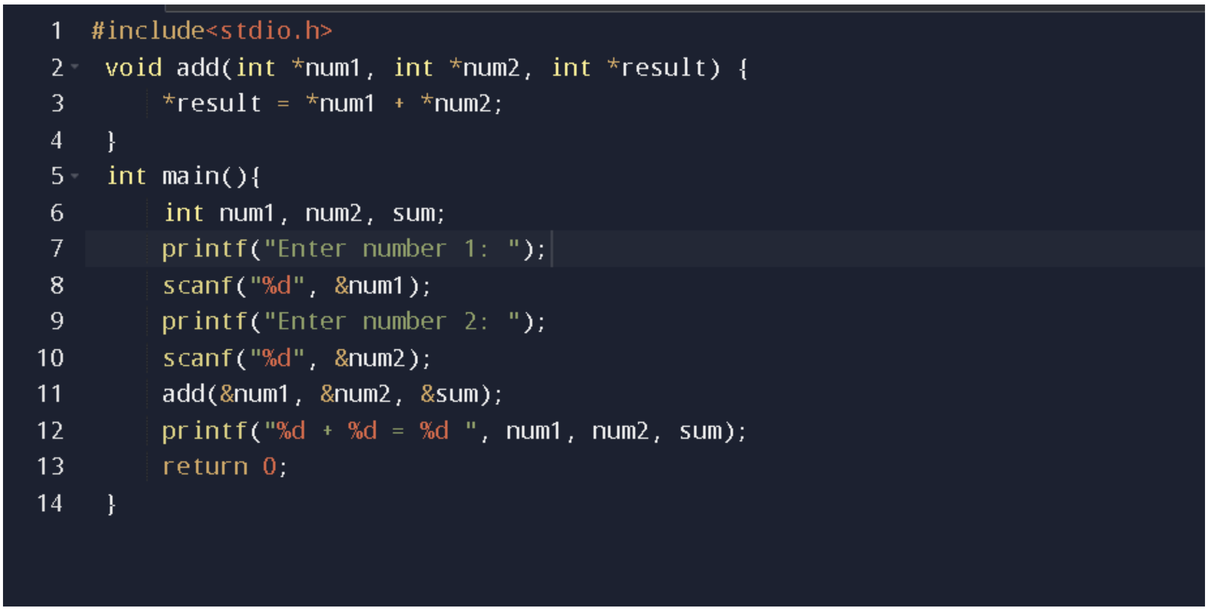
```c
#include<stdio.h>

int findMax(int *num1, int *num2){
    if (*num1 > *num2){
        return *num1;
    } else{
        return *num2;
    }
}
int main(){
    int num1, num2, max;
    printf("Number 1: ");
    scanf("%d", &num1);
    printf("Number 2: ");
    scanf("%d", &num2);
    max = findMax(&num1, &num2);
    printf("The maximum between %d and %d is : %d\n", num1, num2, max);
    return 0;
}

```



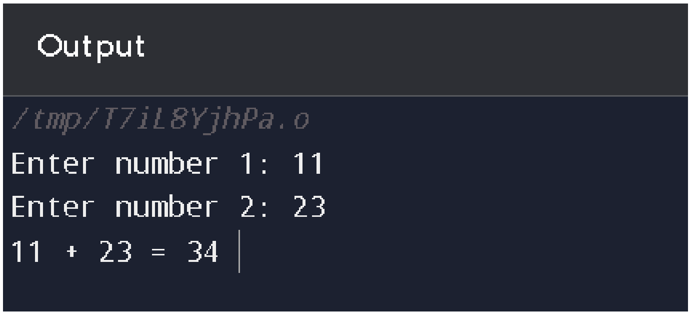
Output

```
/tmp/T7iL8YjhPa.o
Number 1: 11
Number 2: 23
The maximum between 11 and 23 is : 23
```

6. Write a program in C to add numbers using call by reference.



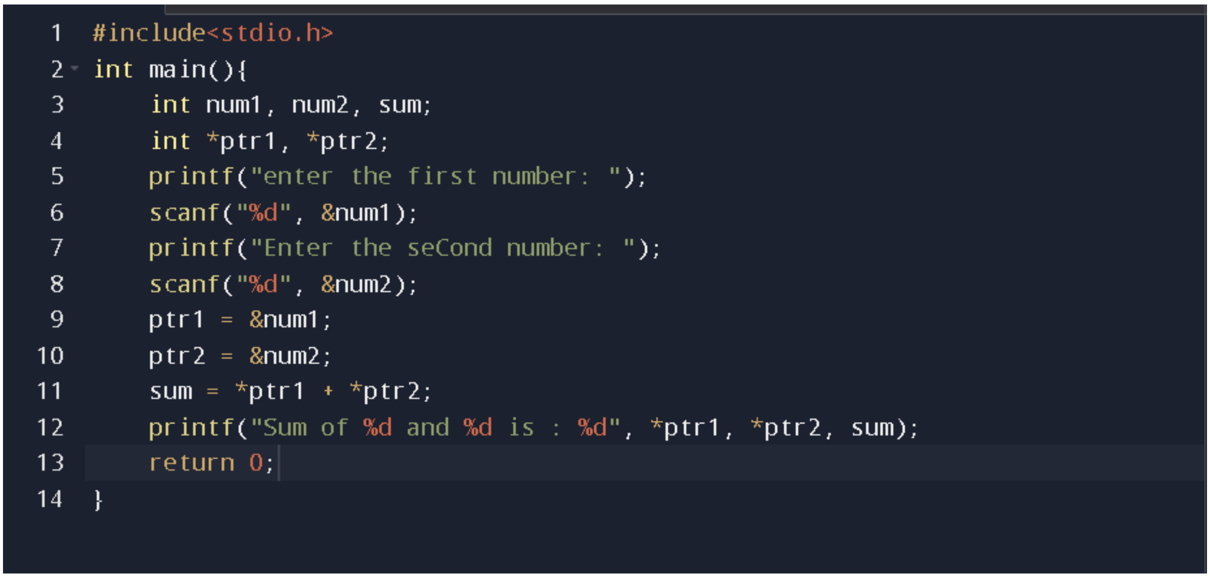
```
#include<stdio.h>
void add(int *num1, int *num2, int *result) {
    *result = *num1 + *num2;
}
int main(){
    int num1, num2, sum;
    printf("Enter number 1: ");
    scanf("%d", &num1);
    printf("Enter number 2: ");
    scanf("%d", &num2);
    add(&num1, &num2, &sum);
    printf("%d + %d = %d ", num1, num2, sum);
    return 0;
}
```



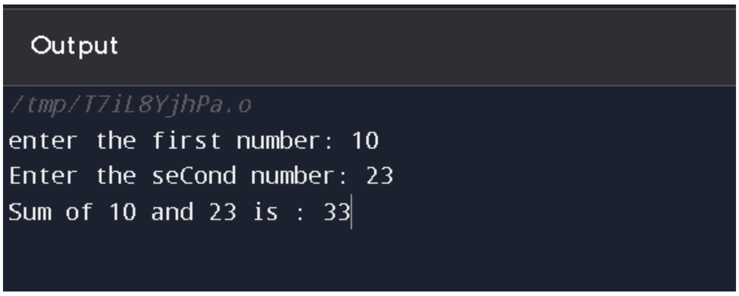
Output

```
/tmp/T7iL8YjhPa.o
Enter number 1: 11
Enter number 2: 23
11 + 23 = 34
```

7. Write a program in C to add two numbers using pointers.

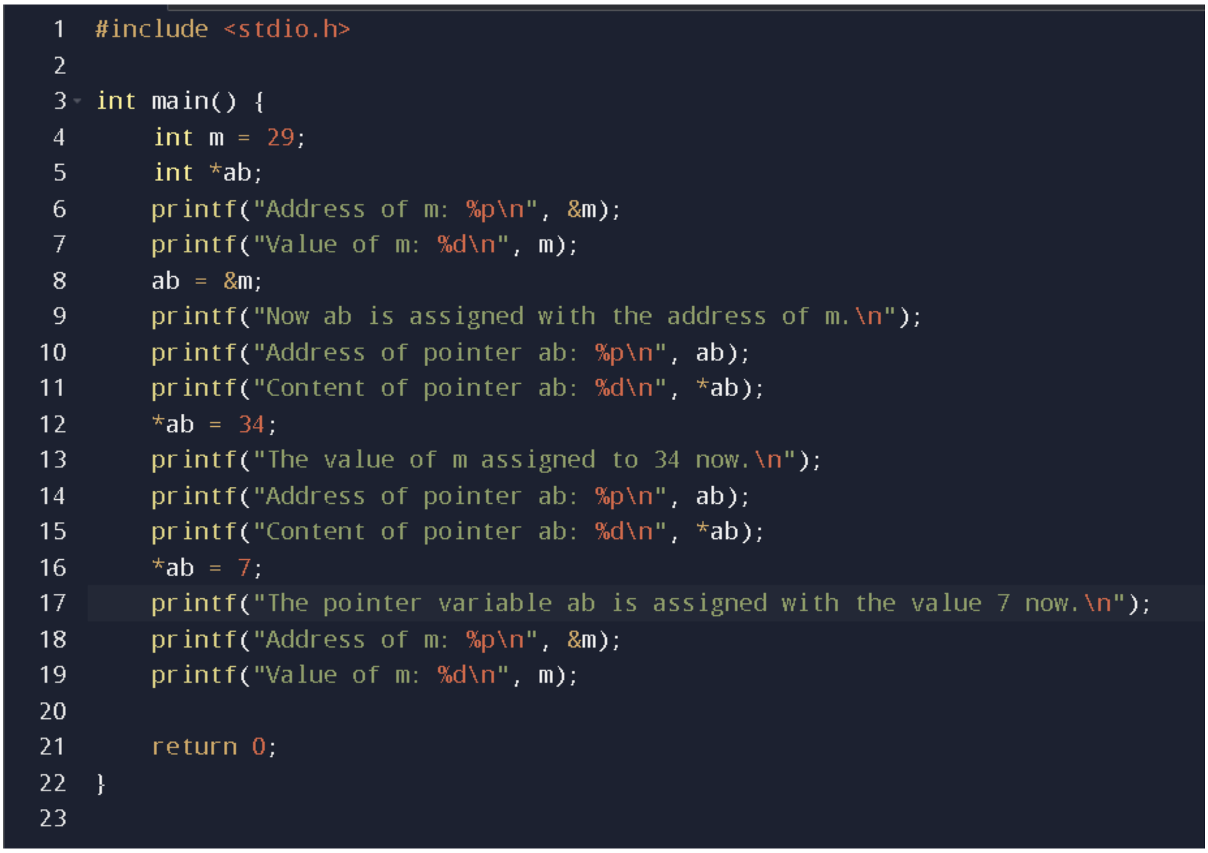


```
#include<stdio.h>
int main(){
    int num1, num2, sum;
    int *ptr1, *ptr2;
    printf("enter the first number: ");
    scanf("%d", &num1);
    printf("Enter the seCond number: ");
    scanf("%d", &num2);
    ptr1 = &num1;
    ptr2 = &num2;
    sum = *ptr1 + *ptr2;
    printf("Sum of %d and %d is : %d", *ptr1, *ptr2, sum);
    return 0;
}
```

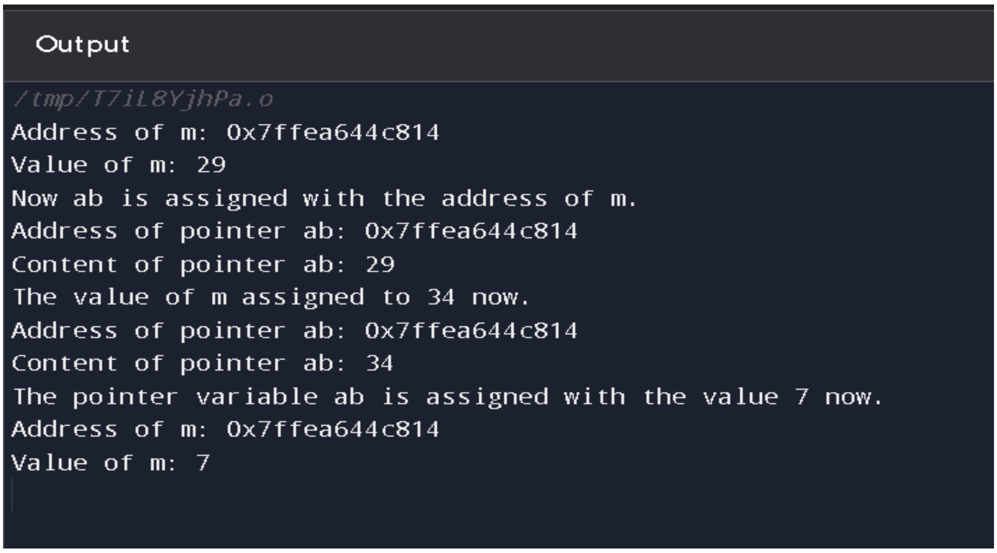


```
/tmp/T7iL8YjhPa.o
enter the first number: 10
Enter the seCond number: 23
Sum of 10 and 23 is : 33
```

8. Write a program in C to demonstrate how to handle pointers in a program.



```c
#include <stdio.h>

int main() {
    int m = 29;
    int *ab;
    printf("Address of m: %p\n", &m);
    printf("Value of m: %d\n", m);
    ab = &m;
    printf("Now ab is assigned with the address of m.\n");
    printf("Address of pointer ab: %p\n", ab);
    printf("Content of pointer ab: %d\n", *ab);
    *ab = 34;
    printf("The value of m assigned to 34 now.\n");
    printf("Address of pointer ab: %p\n", ab);
    printf("Content of pointer ab: %d\n", *ab);
    *ab = 7;
    printf("The pointer variable ab is assigned with the value 7 now.\n");
    printf("Address of m: %p\n", &m);
    printf("Value of m: %d\n", m);

    return 0;
}

```



```
/tmp/T7iL8YjhPa.o
Address of m: 0x7ffea644c814
Value of m: 29
Now ab is assigned with the address of m.
Address of pointer ab: 0x7ffea644c814
Content of pointer ab: 29
The value of m assigned to 34 now.
Address of pointer ab: 0x7ffea644c814
Content of pointer ab: 34
The pointer variable ab is assigned with the value 7 now.
Address of m: 0x7ffea644c814
Value of m: 7
```