**Program 1:Swapping two numbers**

#include<stdio.h>

int main()

{

int a;

int b;

int temp;

int \*ptr1, \*ptr2;

printf("Enter value of a:");

scanf("%d", &a);

printf("Enter value of b:");

scanf("%d", &b);

printf(" Before Swapping two numbers is a =%d and b =%d", a, b,"\n");

ptr1=&a;

ptr2=&b;

temp=\*ptr1;

\*ptr1=\*ptr2;

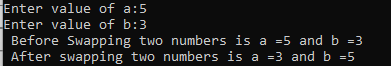
\*ptr2=temp;

printf("\n After swapping two numbers is a =%d and b =%d", a, b);

return 0;

}

**OUTPUT:**



**Program 2:Dyamic Memory Allocation**

#include <stdio.h>

#include <stdlib.h>

int main()

{ int\* ptr;

int n, i;

n = 5;

printf("Enter number of elements: %d\n", n);

ptr=(int\*)malloc(n\*sizeof(int));

if (ptr == NULL) {

printf("Memory not allocated.\n");

exit(0);

}

else {

printf("Memory successfully allocated using malloc.\n");

for (i = 0; i < n; ++i) {

ptr[i] = i + 1;

}

printf("The elements of the array are: ");

for (i = 0; i < n; ++i) {

printf("%d, ", ptr[i]);

}

ptr = (int\*)calloc(n, sizeof(int));

if (ptr == NULL) {

printf("Memory not allocated.\n");

exit(0);

}

else {

printf("Memory successfully allocated using calloc.\n");

for (i = 0; i < n; ++i) {

ptr[i] = i + 1;

}

printf("The elements of the array are: ");

for (i = 0; i < n; ++i) {

printf("%d, ", ptr[i]);

}

n = 10;

printf("\n\nEnter the new size of the array: %d\n", n);

ptr = (int\*)realloc(ptr, n \* sizeof(int));

printf("Memory successfully re-allocated using realloc.\n");

for (i = 5; i < n; ++i) {

ptr[i] = i + 1;

}

printf("The elements of the array are: ");

for (i = 0; i < n; ++i) {

printf("%d, ", ptr[i]);

}

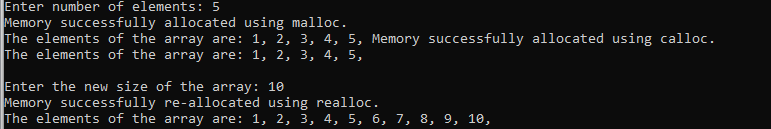
free(ptr);}

return 0;

}

}

**OUTPUT:**



**Program 3:Stack Implementation**

#define SIZE 5

int stack[SIZE];

int top = -1;

void push(int element);

void pop();

void display();

int main() {

int choice, element;

do {

printf("\nStack Operations:\n");

printf("1. Push\n");

printf("2. Pop\n");

printf("3. Display\n");

printf("4. Exit\n");

printf("Enter your choice: ");

scanf("%d", &choice);

switch (choice) {

case 1:

printf("Enter the element to push: ");

scanf("%d", &element);

push(element);

break;

case 2:

pop();

break;

case 3:

display();

break;

case 4:

printf("Exiting the program.\n");

break;

default:

printf("Invalid choice. Please enter a valid option.\n");

}

} while (choice != 4);

return 0;

}

void push(int element) {

if (top == SIZE - 1) {

printf("Stack Overflow! Cannot push element.\n");

} else {

top++;

stack[top] = element;

printf("%d pushed onto the stack.\n", element);

}

}

**void pop() {**

**if (top == -1) {**

**printf("Stack Underflow! Cannot pop element.\n");**

**} else {**

**printf("%d popped from the stack.\n", stack[top]);**

top--;

}

}

void display() {

if (top == -1) {

printf("Stack is empty. Nothing to display.\n");

} else {

printf("Elements in the stack:\n");

for (int i = 0; i <= top; i++) {

printf("%d ", stack[i]);

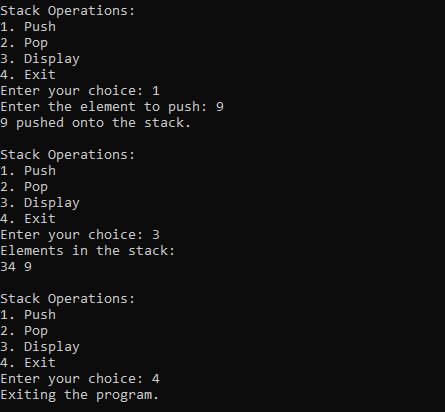
}

printf("\n");

}

}

**OUTPUT:**

****