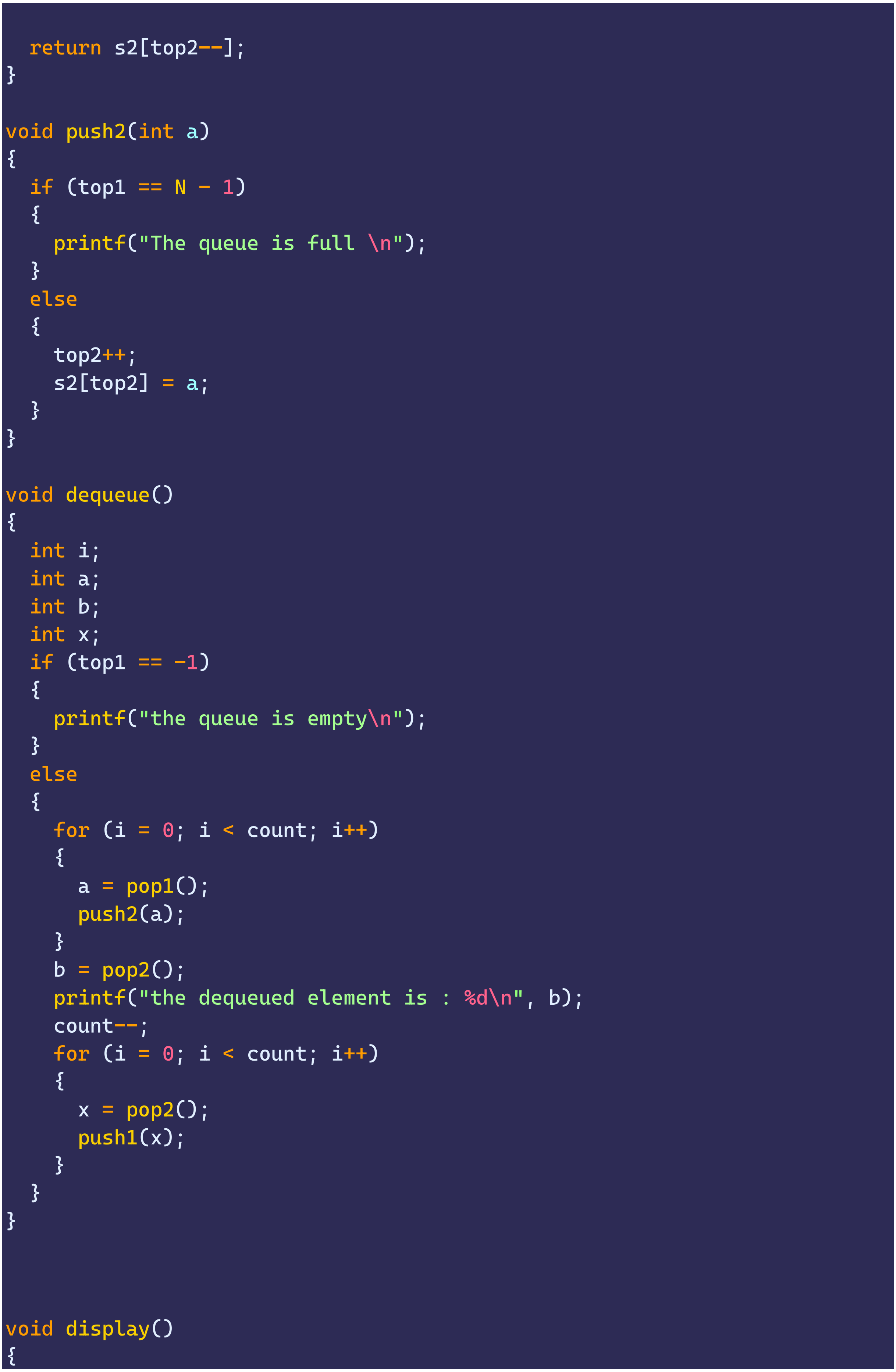
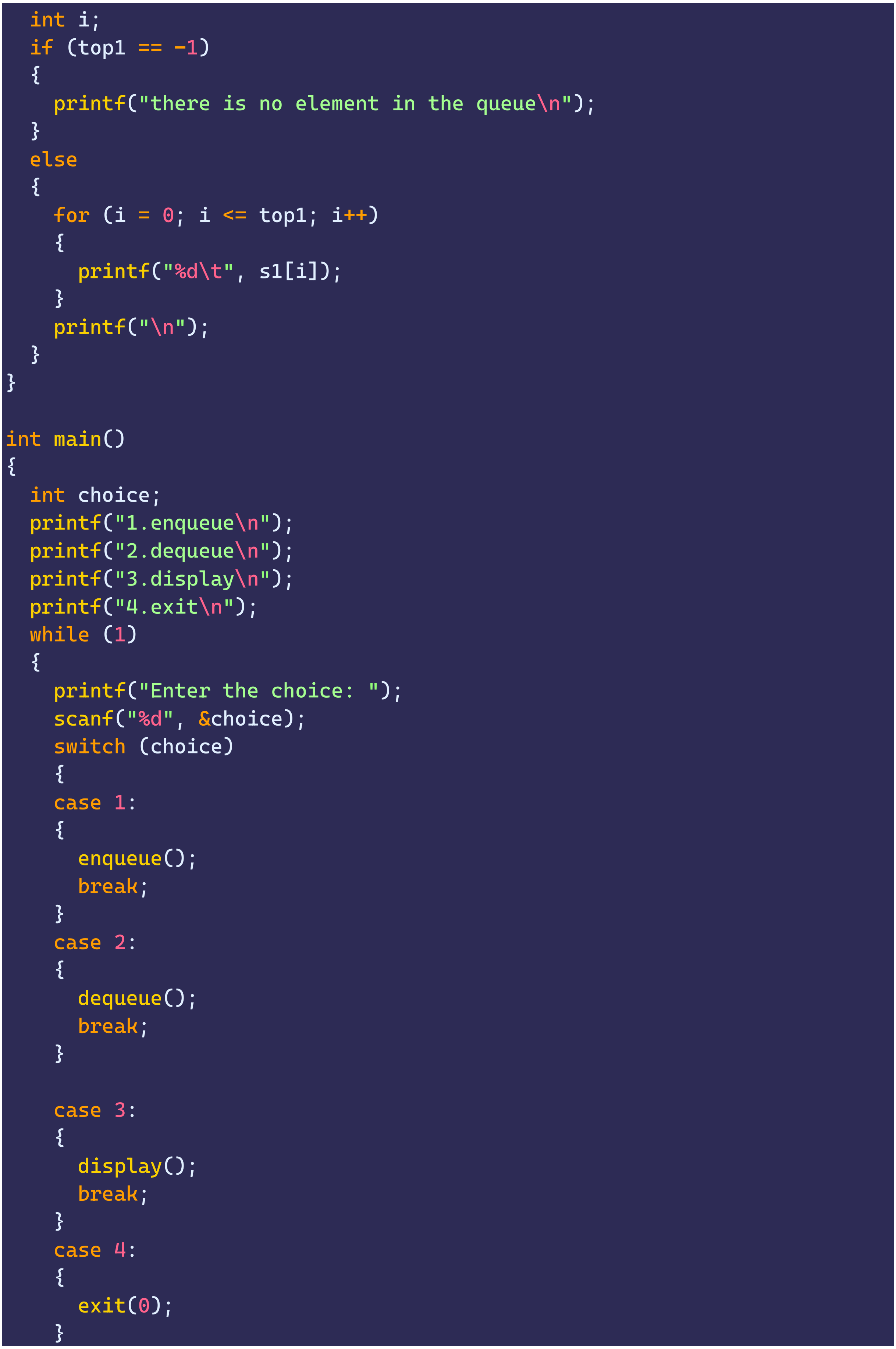
Queue using two stacks

```
#include <stdio.h>
#include <stdlib.h>
#define N 100

int s1[N];
int s2[N];
int top1 = -1;
int top2 = -1;
int count = 0;

void push1(int a)
{
  if (top1 == N - 1)
  {
    printf("The queue is full \n");
  }
  else
  {
    top1++;
    s1[top1] = a;
  }
}

void enqueue()
{
  int data;
  if (top1 == N - 1)
  {
    printf("overflow\n");
  }
  else
  {
    printf("Enter the data: ");
    scanf("%d", &data);
    push1(data);
    count++;
  }
}

int pop1()
{
  return s1[top1--];
}

int pop2()
{
```



```c
  return s2[top2--];
}

void push2(int a)
{
  if (top1 == N - 1)
  {
    printf("The queue is full \n");
  }
  else
  {
    top2++;
    s2[top2] = a;
  }
}

void dequeue()
{
  int i;
  int a;
  int b;
  int x;
  if (top1 == -1)
  {
    printf("the queue is empty\n");
  }
  else
  {
    for (i = 0; i < count; i++)
    {
      a = pop1();
      push2(a);
    }
    b = pop2();
    printf("the dequeued element is : %d\n", b);
    count--;
    for (i = 0; i < count; i++)
    {
      x = pop2();
      push1(x);
    }
  }
}



void display()
{
```



```
  int i;
  if (top1 == -1)
  {
    printf("there is no element in the queue\n");
  }
  else
  {
    for (i = 0; i <= top1; i++)
    {
      printf("%d\t", s1[i]);
    }
    printf("\n");
  }
}

int main()
{
  int choice;
  printf("1.enqueue\n");
  printf("2.dequeue\n");
  printf("3.display\n");
  printf("4.exit\n");
  while (1)
  {
    printf("Enter the choice: ");
    scanf("%d", &choice);
    switch (choice)
    {
    case 1:
    {
      enqueue();
      break;
    }
    case 2:
    {
      dequeue();
      break;
    }

    case 3:
    {
      display();
      break;
    }
    case 4:
    {
      exit(0);
    }
    default:
    {
      printf("invalid choice\n");
    }
    }
  }
  return 0;
}
```