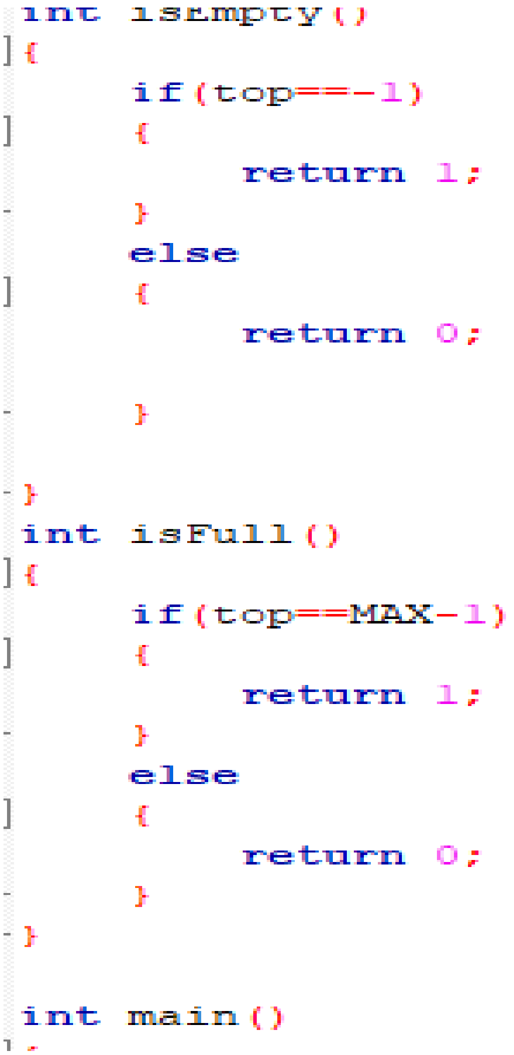
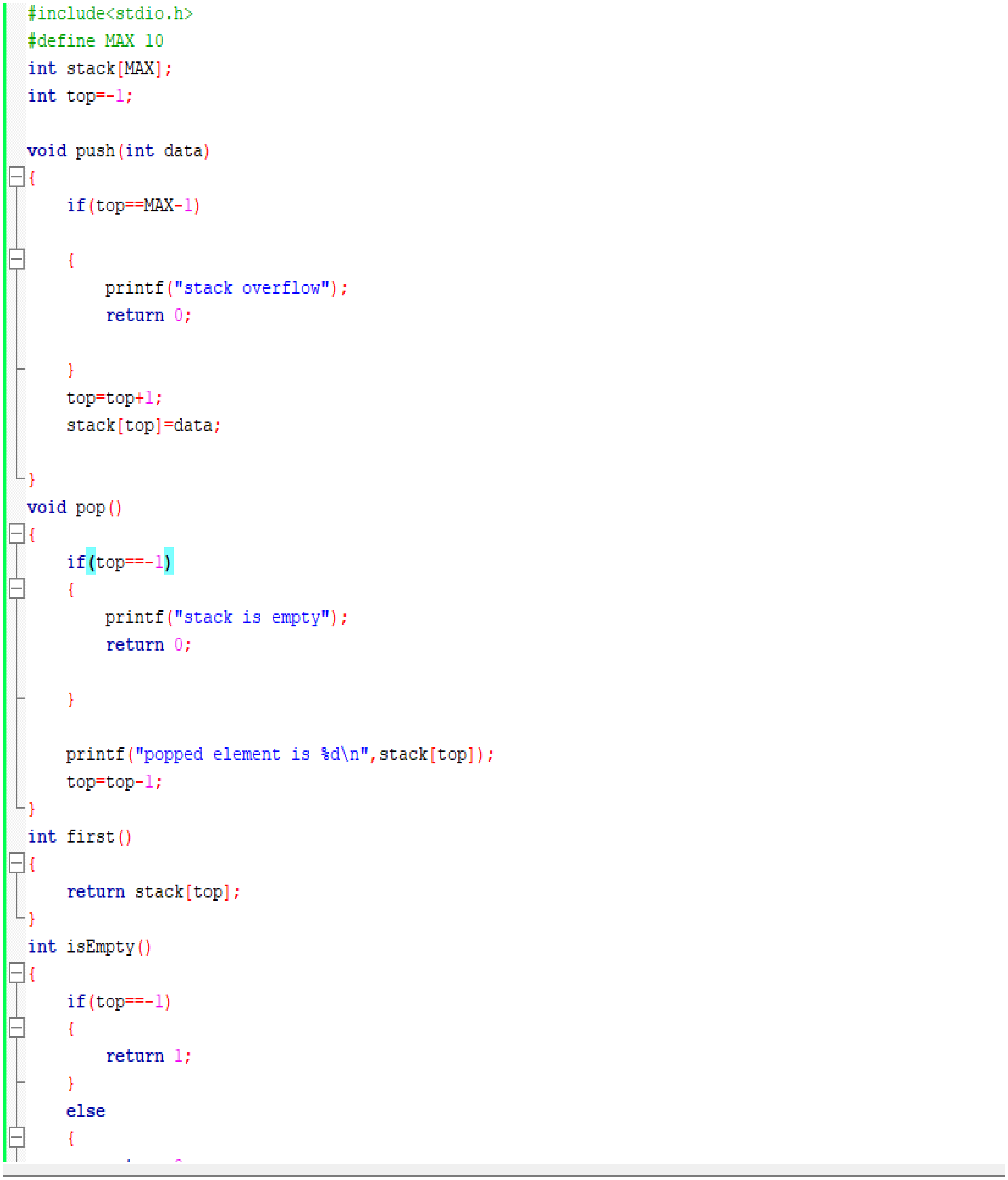
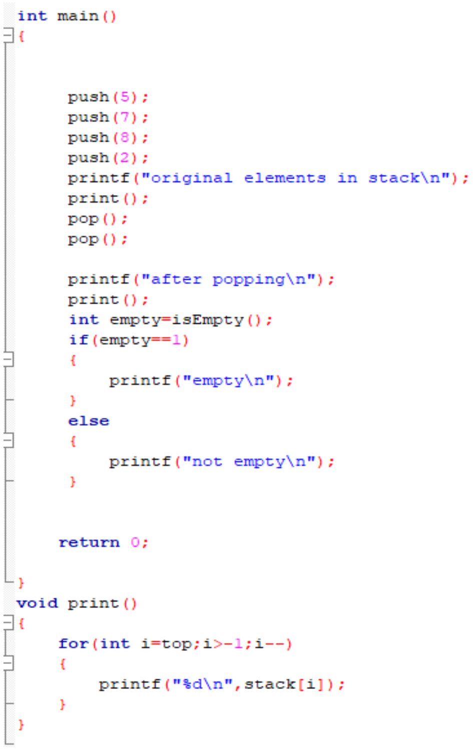


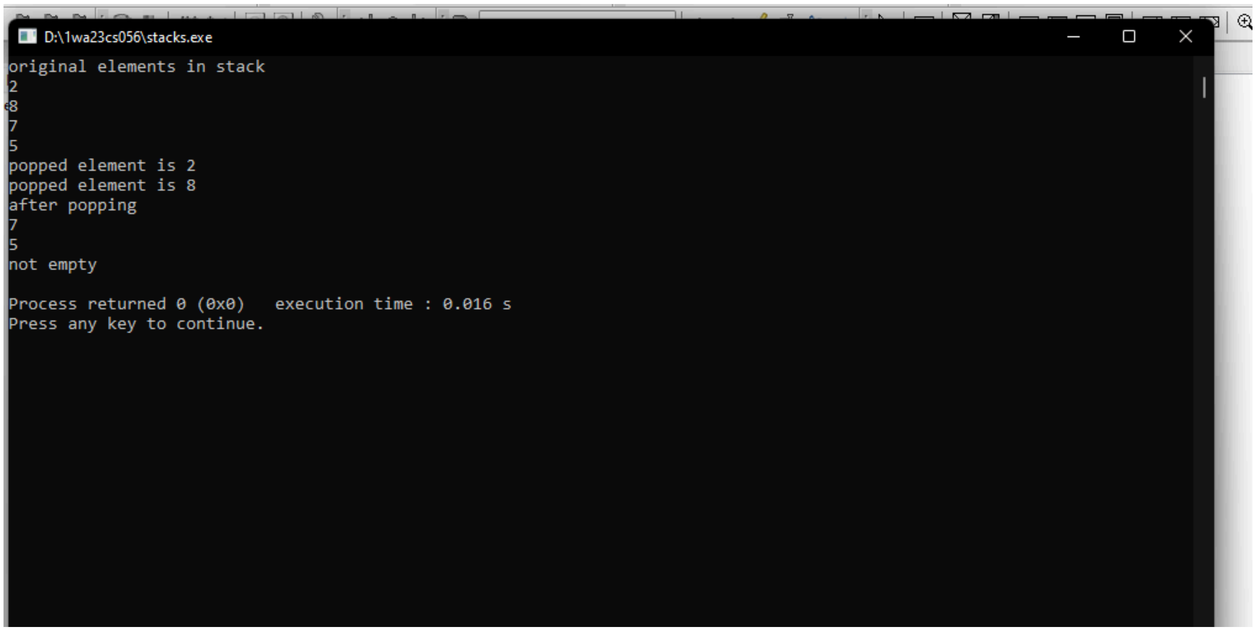
```
int isEmpty()
{
    if(top==-1)
    {
        return 1;
    }
    else
    {
        return 0;

    }

}
int isFull()
{
    if(top==MAX-1)
    {
        return 1;
    }
    else
    {
        return 0;
    }
}

int main()
{
```



```
#include<stdio.h>
#define MAX 10
int stack[MAX];
int top=-1;

void push(int data)
{
    if(top==MAX-1)

    {
        printf("stack overflow");
        return 0;

    }
    top=top+1;
    stack[top]=data;

}
void pop()
{
    if(top==-1)
    {
        printf("stack is empty");
        return 0;

    }

    printf("popped element is %d\n",stack[top]);
    top=top-1;
}
int first()
{
    return stack[top];
}
int isEmpty()
{
    if(top==-1)
    {
        return 1;
    }
    else
    {
```



```
int main()
{


     push(5);
     push(7);
     push(8);
     push(2);
     printf("original elements in stack\n");
     print();
     pop();
     pop();

     printf("after popping\n");
     print();
     int empty=isEmpty();
     if(empty==1)
     {
         printf("empty\n");
     }
     else
     {
         printf("not empty\n");
     }


    return 0;

}
void print()
{
    for(int i=top;i>-1;i--)
    {
        printf("%d\n",stack[i]);
    }
}
```



D:\1wa23cs056\stacks.exe

```
original elements in stack
2
8
7
5
popped element is 2
popped element is 8
after popping
7
5
not empty

Process returned 0 (0x0)   execution time : 0.016 s
Press any key to continue.
```