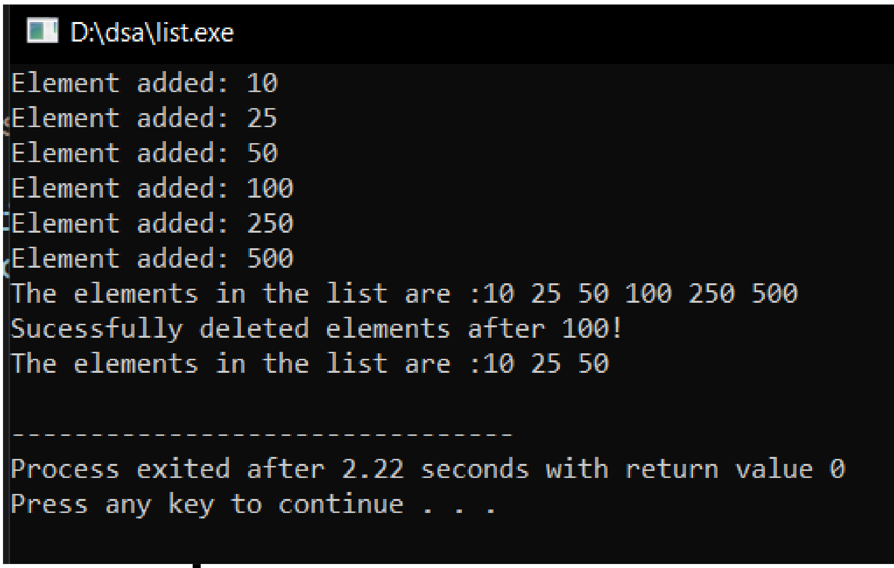
```
#include <stdio.h>
#include <stdlib.h>
#define MAXNODE 10
struct nodetype{
   int info, next;
};
struct nodetype node[MAXNODE];
int avail = 0, i;

void initlist(){
   int i;
   for (i = 0; i < MAXNODE-1; i++){
      node[i].next = i + 1;
   }
   node[MAXNODE - 1].next = -1;
}
int getnode(){
   int p;
   if (avail == -1){
      printf("Data can't be added \n");
      exit(0);
   }
   p = avail;
   avail = node[avail].next;
   return p;
}
void freenode(int p){
   node[p].next = avail;
   avail = p;
}
void insafter(int p, int q){
    int r;
   if (p == -1){
      printf("Invalid operation");
      exit(0);
   }
   r = getnode();
   node[r].info = q;
   node[r].next = node[p].next;
   node[p].next = r;
   printf("Element added: %d\n", q);
}
void delafter(int p){
   int q, r;
   if (p == -1 || node[p].next == -1){
      printf("Invalid deletion");
      exit(0);
   }
   q = node[p].next;
   r = node[q].info;
   node[p].next = node[q].next;
   freenode(q);
   printf("Sucessfully deleted elements after %d!\n", r);
}
```

```
void display(){
 printf("The elements in the list are :");
 int i;
 for (i = 0; i < avail; i++){
 printf("%d ", node[i].info);
 }
 printf("\n");
}
int main(){
 initlist();
 insafter(0, 10);
 insafter(1, 25);
 insafter(2, 50);
 insafter(3, 100);
 insafter(4, 250);
 insafter(5, 500);
 display();
 delafter(3);
 display();
}
```

// OUTPUT



D:\dsa\list.exe

```
Element added: 10
Element added: 25
Element added: 50
Element added: 100
Element added: 250
Element added: 500
The elements in the list are :10 25 50 100 250 500
Sucessfully deleted elements after 100!
The elements in the list are :10 25 50

--------------------------------
Process exited after 2.22 seconds with return value 0
Press any key to continue . . .
```