```c
#include<stdio.h>
#include<process.h>
#include<stdlib.h>
#include<string.h>

struct NODE
{
 int info;
 struct NODE*link;
};
typedef struct NODE*node;
node getnode()
{
 node x;
 x=(node)malloc(sizeof(struct NODE));
 if(x==NULL)
 {
  printf("Out of Memory\n");
  exit(0);
 }
 return x;
}



node ins_front(node first,int item)
{
 node temp;
 temp=getnode();
 temp->info=item;
 temp->link=first;
 return temp;
}



node extract(char *s,node head)
{
```

```
36
37  node extract(char *s,node head)
38  {
39   int n,i;
40   for(i=0;i<strlen(s);i++)
41   {
42    n=s[i]-'0';
43      head=ins_front(head,n);
44   }
45      return head;
46  }
47
48
49  node addlong(node head1,node head2,node head3)
50  {
51      int temp,sum,carry=0;
52      node cur1,cur2;
53      cur1=head1;
54      cur2=head2;
55      while(cur1!=NULL&&cur2!=NULL)
56      {
57          temp=cur1->info+cur2->info+carry;
58          if(temp>9)
59          {
60          sum=temp%10;
61          carry=temp/10;
62          }
63          else
64          {
65          sum=temp;
66          carry=0;
67          }
68          head3=ins_front(head3,sum);
69          cur1=cur1->link;
70          cur2=cur2->link;
71      }
72      while(cur1!=NULL)
73      {
```

```
70          cur2=cur2->link;
71      }
72      while(cur1!=NULL)
73      {
74      temp=cur1->info+carry;
75      if(temp>9)
76      {
77      sum=temp%10;
78      carry=temp/10;
79      }
80      else
81      {
82      sum=temp;
83      carry=0;
84      }
85      head3=ins_front(head3,sum);
86      cur1=cur1->link;
87      }
88    while(cur2!=NULL)
89      {
90      temp=cur2->info+carry;
91      if(temp>9)
92      {
93      sum=temp%10;
94      carry=temp/10;
95      }
96      else
97      {
98      sum=temp;
99      carry=0;
100     }
101     head3=ins_front(head3,sum);
102     cur2=cur2->link;
103     }
104
105     if(cur1==NULL&&cur2==NULL)
106     {
107         if(carry==1)
```

```
    carry=0;
    }
    head3=ins_front(head3,sum);
    cur2=cur2->link;
    }

    if(cur1==NULL&&cur2==NULL)
    {
        if(carry==1)
        head3=ins_front(head3,carry);
    }
    return head3;
  }



void display(node first)
{
node cur;
if(first==NULL)
{
printf("Empty\n");
return;
}
cur=first;
while(cur!=NULL)
{
printf("%d\t",cur->info);
cur=cur->link;
}
}
int main()
{

    node head1=NULL;
    node head2=NULL;
    node head3=NULL;
    char s1[30],s2[30];
```

```
118 if(first==NULL)
119 {
120 printf("Empty\n");
121 return;
122 }
123 cur=first;
124 while(cur!=NULL)
125 {
126 printf("%d\t",cur->info);
127 cur=cur->link;
128 }
129 }
130 int main()
131 {
132
133     node head1=NULL;
134     node head2=NULL;
135     node head3=NULL;
136     char s1[30],s2[30];
137     printf("\nEnter first integer\n");
138     scanf("%s",s1);
139     head1=extract(s1,head1);
140     display(head1);
141     printf("\nEnter second integer\n");
142     scanf("%s",s2);
143     head2=extract(s2,head2);
144     display(head2);
145     head3=addlong(head1,head2,head3);
146     printf("\nThe result is\n");
147     display(head3);
148     return 0;
149 }
150
151
152
153
154
155
```


C:\WINDOWS\SYSTEM32\cmd.exe
Enter first integer
123456789
9 8 7 6 5 4 3 2 1
Enter second integer
103456789
9 8 7 6 5 4 3 0 1
The result is
2 2 6 9 1 3 5 7 8
(program exited with code: 0)
Press any key to continue . . .