Question 1:

Algorithm to traverse a linked list and delete the negative nodes.

Algorithm :: Delete(start)

1. if (start==NULL),then

1.1 Display "Underflow"

1.2 Stop and exit

2. Set point=start

3. Set prev=start

4. Repeat till 4.2.2 while(point!=NULL)

4.1 if (point->data<0), then

4.1.1 if (point=start), then

4.1.1.1 if (point->next = NULL), then

4.1.1.1.1 Set start=NULL

4.1.1.2 Else

4.1.1.2.1 Set start=point->next

4.1.2 Else if (point->next=NULL), then

4.1.2.1 Set prev->next=NULL

4.1.3 Else

4.1.3.1 Set prev->next=point->next

4.1.4 Set point=start

4.2 Else

4.2.1 Set prev=point

4.2.2 Set point=point->next

5. Stop and exit

Question 2:

Program to convert an infix expression to a postfix expression

#include<stdio.h>

char arr[100];

char stack[100];

char post[100];

main()

{

printf("Enter infix\n");

gets(arr);

int i=0;

while(arr[i]!='\0')

i++;

arr[i]=')';

stack[0]='(';

int k=-1,l=0;i=0;

while(arr[i]!='\0')

{

if(arr[i]=='^'||arr[i]=='+'||arr[i]=='/'||arr[i]=='\*'||arr[i]=='-'||arr[i]=='%')

{

if(arr[i]=='+'||arr[i]=='-')

while(stack[l]!='(')

{

k++;

post[k]=stack[l];

l--;

}

else if(arr[i]=='\*'||arr[i]=='/'|| arr[i]=='%')

while(stack[l]!='(' && stack[l]!='+' && stack[l]!='-')

{

k++;

post[k]=stack[l];

l--;

}

else if(arr[i]=='^')

while(stack[l]!='(' && stack[l]!='+' && stack[l]!='-'&& stack[l]!='\*'

&& stack[l]!='/' && stack[l]!='%')

{

k++;

post[k]=stack[l];

l--;

}

l++;

stack[l]=arr[i];

}

else if((arr[i]>64 && arr[i]<91)||(arr[i]>96 && arr[i]<123)||(arr[i]>47

&& arr[i]<58))

{

k++;

post[k]=arr[i];

}

else if(arr[i]=='(')

{

l++;

stack[l]='(';

}

else if(arr[i]==')')

{

while(stack[l]!='(')

{

k++;

post[k]=stack[l];

l--;

}

l--;

}

i++;

}

puts(post);

}

Name: Asif Ansari

Roll No: 1205268

Section : CS-4

Branch: Computer Science and Engineering