

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
#define MAX 20
```

```
char stack[MAX];
int top = -1;
char pop();
void push(char item);
```

```
int prcd(char symbol)
{
    switch(symbol)
    {
        case '+':
        case '-':
            return 2;
        case '*':
        case '/':
            return 4;
        case '^':
        case '$':
            return 6;
        case '(':
        case ')':
        case '#':
            return 1;
    }
}
```

```
int isoperator(char symbol)
{
    switch(symbol)
    {
        case '+':
        case '-':
        case '*':
        case '/':
        case '^':
        case '$':
        case '(':
        case ')':
            return 1;
        default:
            return 0;
    }
}
```

```
void convertip(char infix[],char prefix[])
{
    int i,symbol,j=0;
```

```

char test[MAX];

infix=strrev(infix);
stack[++top]='#';

for(i=0;i<strlen(infix);i++)
{
    symbol=infix[i];
    if(isoperator(symbol)==0)
    {
        prefix[j]=symbol;
        j++;
    }
    else
    {
        if(symbol==' ')
        {
            push(symbol);
        }
        else if(symbol=='(')
        {
            while(stack[top]!=' ')
            {
                prefix[j]=pop();
                j++;
            }
            pop();//pop out (
        }
        else
        {
            if(prcd(symbol)>prcd(stack[top]))
            {
                push(symbol);
            }
            else
            {
                while(prcd(symbol)<=prcd(stack[top]))
                {
                    prefix[j]=pop();
                    j++;
                }
                push(symbol);
            }
        }
    }
}

while(stack[top]!='#')
{
    prefix[j]=pop();
    j++;
}

```

```
    }
    prefix[j]='\0';//null terminate string.
    prefix=strrev(prefix);
}
```

```
int main()
{
    char infix[20],prefix[20];

    printf("Masukkan aritmatika infix :");
    gets(infix);
    convertip(infix,prefix);
    printf(" Hasil Prefixnya :\n");
    puts(prefix);
    getch();

    return 0;
}
```

```
void push(char item)
{
    top++;
    stack[top]=item;
}
```

```
char pop()
{
    char a;
    a=stack[top];
    top--;
    return a;
}
```