Program 2 (Infix To Postfix)

```
int priority(char ch)
{
  Switch(ch) {
        case '+' :
        case '-' : return 1;
        case '*' :
        case '/' : return 2;
        case '^' : return 3;
        default : return 0;
    }
}

// conversion code
while (infix[i] != '\0') {
      switch(infix[i]) {
            case '(' : push(infix[i]);
                       break;
            case ')' : while((item = pop())!='(')
                       {
                           printf("%c",item);
                       }
                       break;
            case '+' :
            case '-' :
            case '*' :
            case '/' :
            case '^' :
```

```
            while (!stackempty() &&
                            priority(infix[i]) <=
                                     priority(stacktop()
             {
              item = pop();
              printf("%c", item);
             }
              push(infix[i]);
              break;
        default : printf("%c",infix[i]);
                  break;
            }
            i++;
     }
     while (!stackempty()) {
            printf("%c", pop());
     }
     return 0;
   }
```