Infix to Postfix Pseudocode

Carrano, 4th edition, pp. 310-311

```
for (each character ch in the infix expression){
  switch(ch){
    case operand:    // append operand to end of PE    操作数就直接进结果字符串
      postfixExp = postfixExp + ch
      break
    case '(':        // save '(' on stack     (就继续进栈
      aStack.push(ch)
      break
    case ')':        // pop stack until matching '('
      while (top of stack is not '('){            遇到）就pop直到遇到（
        postfixExp = postfixExp + (top of aStack)
        aStack.pop()
      }  // end while
      aStack.pop()   // remove the '('
      break
    case operator:   // process stack operators of    如果是操作符
                     // greater precedence
      while (!aStack.isEmpty() and
             top of stack is not '(' and
             precedence(ch) <= precedence(top of aStack)){
        postfixExp = postfixExp + (top of aStack)
        aStack.pop()
      } // end while
      aStack.push(ch)   // save new operator
      break
  } // end switch
} // end for
// append to postfixExp the operators remaining on the stack
while(!aStack.isEmpty()){
  postfixExp = postfixExp + (top of aStack)
  aStack.pop()
} // end while
```