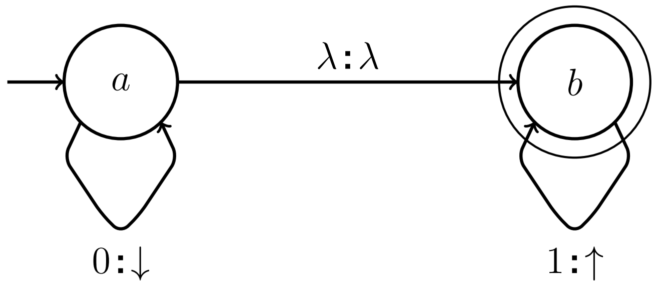
Programming Dictionary

By Charles Cook

Begun on Dec. 6th, 2018

1 Compiler Theoretical Foundations

Push Down Automata for strings of the form 0^n1^n



Key:

Symbol	Meaning
a, b	State Node
0, 1	Symbol to Print
$\downarrow$	Push (onto the stack)
$\uparrow$	Pull (off of the stack)
λ	Null operation (no print, push, or pull)

2 Programming Tools & Languages: First Ten Questions

2.1 Swap Function

2.1.1 Pseudocode for *swap*

```
void swap(a, b) {
        temp = a;
        a = b;
        b = temp;
}

void swapNoTemp(a, b) {
        a += b; // a = a + b
        b -= a; // b = b - (a + b) = -a
        b *= -1; // b = a
        a -= b; // a = a + b - a = b
}
```

2.1.2 *swaptest.c*

```
#include <stdio.h>
void swap(int *a, int *b) {
        *a += *b;
        *b -= *a;
        *b *= -1;
        *a -= *b;
}

int main() {
        int x, y, z;
        x = 10;
        y = 13;
        z = 2;

        swap(&x, &y);
        swap(&x, &z);
        swap(&y, &z);
        printf("x: &d\ny: &d\nz: &d\n", x, y, z);

        return 0;
}
```

2.1.3 *swaptest.cpp*

```
#include <iostream>
using namespace std;

void swap(int *a, int *b) {
        *a += *b;
        *b -= *a;
        *b *= -1;
        *a -= *b;
}

int main() {
        int x, y;
        x = 13;
        y = 29;

        cout << x << ", " << y << "\n";
        swap(x, y);
        cout << x << ", " << y << "\n";

        return 0;
}
```

2.2 Reverse an array with no extra space (pseudocode)

```
void reverseArray(array, int length) {
        for (int i = 0; i < length / 2; i++) {
                swap(array[i], array[length - i - 1]);
        }
}
```

2.3 Reverse a doubly linked list (pseudocode)

```
struct Node {
        int data;
        struct Node *next;
        struct Node *prev;
};

void reverseDLL(head, tail) {
        struct Node tempH = head;
        struct Node tempT = tail;

        while (tempH -> next != tempT -> prev && tempH != tempT) {
                swap(tempH -> data, tempT -> data);
        }
}
```

2.4 Reverse a doubly linked list recursively (pseudocode)

```
void reverseDLL_Recursive(head, tail) {
        if (head -> next != tail -> prev && head != tail) {
                swap(head -> data, tail -> data);
                reverseDLL_Recursive(head -> next, tail -> prev);
        }
}
```