# Introduction to Maths, Flowcharts, and Pseudocode - Algorithms (MATH1812)[1]

John S Butler (TU Dublin)

## Mathematics - Single Loop

### Mathematics Example 1

The mathematical expression

$$\sum_{i=0}^{6} 3i,$$

can be expanded as,

$$\sum_{i=0}^{6} 3i = 3(0)+3(1)+3(2)+3(3)+3(4)+3(5)+3(6),$$

or can be written in tabular form:

| $i$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|---|
| $3i$ | $3(0)$ | $3(1)$ | $3(2)$ | $3(3)$ | $3(4)$ | $3(5)$ | $3(6)$ |

where the bottom row is summed giving,

$$\sum_{i=0}^{6} 3i = 63.$$

### Mathematics Example 2

The mathematical expression

$$10 + \sum_{j=-3}^{2} (2+\frac{j}{2}),$$

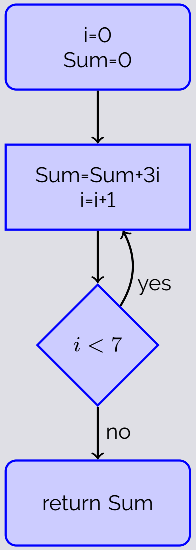
can be expanded as,

$$10+\sum_{j=-3}^{2} (2+\frac{j}{2}) = 10+(2+\frac{-3}{2})+(2+\frac{-2}{2})+(2+\frac{-1}{2})$$
$$+(2+\frac{0}{2})+(2+\frac{1}{2})+(2+\frac{2}{2}),$$

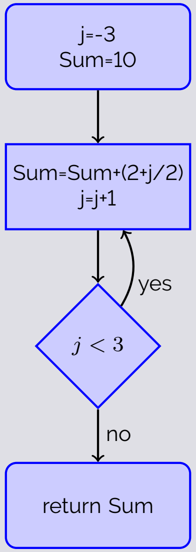
$$10+\sum_{j=-3}^{2} (2+\frac{j}{2}) = 20.5.$$

## Flowchart - Single Loop

### Flowchart Example 1



### Flowchart Example 2



## Pseudocode - Single Loop

### Pseudocode Example 1

Python Pseudocode

```
# Setting up the initial Sum value
Sum=0

# For loop from 0 to 6 with steps of 1
for i in range(0,7):
    Sum=Sum+3*i

return Sum
```

The line by line output of the code for Example 1 is:

| Loop count | i | Sum |
|---|---|---|
| 0 | 0 | 0+3(0)=0 |
| 1 | 1 | 0+3(1)=3 |
| 2 | 2 | 3+3(2)=9 |
| 3 | 3 | 9+3(3)=18 |
| 4 | 4 | 18+3(4)=30 |
| 5 | 5 | 30+3(5)=45 |
| 6 | 6 | 45+3(6)=63 |
| | | 63 |

### Pseudocode Example 2

Python Pseudocode

```
# Setting up the initial Sum value as 10
Sum=10

# For loop from -3 to 2 with steps of 1
for j in range(-3,3):
    Sum=Sum+(2+j/2)

return Sum
```

The line by line output of the code for Example 2 is:

| Loop count | j | Sum |
|---|---|---|
| 0 | -3 | 10+(2-3/2)=10.5 |
| 1 | -2 | 10.5+(2-2/2)=11.5 |
| 2 | -1 | 11.5+(2-1/2)=13 |
| 3 | 0 | 13+(2-0/2)=15.0 |
| 4 | 1 | 15+(2+1/2)=17.5 |
| 5 | 2 | 17.5+(2+2/2)=20.5 |
| | | 20.5 |

[1] Course Website: https://sites.google.com/dit.ie/math1812/home

## Mathematics - Sequential Loops

### Example 3

The mathematical expression

$$-3+\sum_{i=0}^{4}-2i+\sum_{j=-10}^{-6}(j+1)^2,$$

can be expanded as,

$$\begin{gathered}-3+\sum_{i=0}^{4}-2i+\sum_{j=-10}^{-6}(j+1)^2=\\ -3+\\ -2(0)+-2(1)+-2(2)+-2(3)+-2(4)+\\ (-10{+}1)^2{+}(-9{+}1)^2{+}(-8{+}1)^2{+}(-7{+}1)^2{+}(-6{+}1)^2,\end{gathered}$$

written in tabular form:

| $i$ | $0$ | $1$ | $2$ | $3$ | $4$ |
|---|---|---|---|---|---|
| $-2i$ | $-2(0)$ | $-2(1)$ | $-2(2)$ | $-2(3)$ | $-2(4)$ |

where the bottom row is summed giving,

$$\sum_{i=0}^{4}-2i=-20,$$

the second summation is written in tabular form,

| $j$ | $-10$ | $-9$ | $-8$ | $-7$ | $-6$ |
|---|---|---|---|---|---|
| $(j+1)^2$ | $(-9)^2$ | $(-8)^2$ | $(-7)^2$ | $(-6)^2$ | $(-5)^2$ |

summing the bottom row gives,

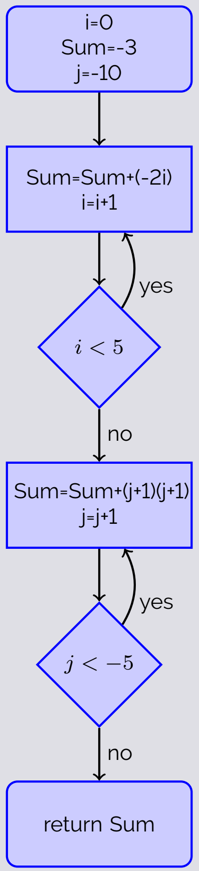
$$\sum_{j=-10}^{-6}(j+1)^2=255,$$

bringing this all together,

$$-3+\sum_{i=0}^{4}-2i+\sum_{j=-10}^{-6}(j+1)^2=-3-20+255=232.$$

## Flowchart - Sequential Loops

### Example 3



## Pseudocode - Sequential Loops

### Example 3

**Python Pseudocode**

```
# Setting up the initial Sum value
Sum=-3

for i in range(0,5):
    Sum=Sum+(-2*i)
    print(Sum)
    print(i)

for j in range(-10,-5):
    Sum=Sum+(j+1)*(j+1)
    print(Sum)
    print(j)

Sum
```

The line by line output of the code for Example 3 is:

| Count | Loop | Sum |
|---|---|---|
| 0 | | -3 |
| | i | -2i |
| 1 | 0 | -3-2(0) |
| 2 | 1 | -3-2(1)=-5 |
| 3 | 2 | -5-2(2)=-9 |
| 4 | 3 | -9-2(3)=-15 |
| 5 | 4 | -15-2(4)=-23 |
| | j | (j+1)(j+1) |
| 6 | -10 | -23+(-10+1)(-10+1)=58 |
| 7 | -9 | 58+(-9+1)(-9+1)=122 |
| 8 | -8 | 122+(-8+1)(-8+1)=171 |
| 9 | -7 | 171+(-7+1)(-7+1)=207 |
| 10 | -6 | 207+(-6+1)(-6+1)=232 |
| | | 232 |

## Mathematics - Double Loop

### Example 4 - Double Loop

The mathematical expression

$$\sum_{i=0}^{3}\sum_{j=0}^{3}(i^2+3j),$$

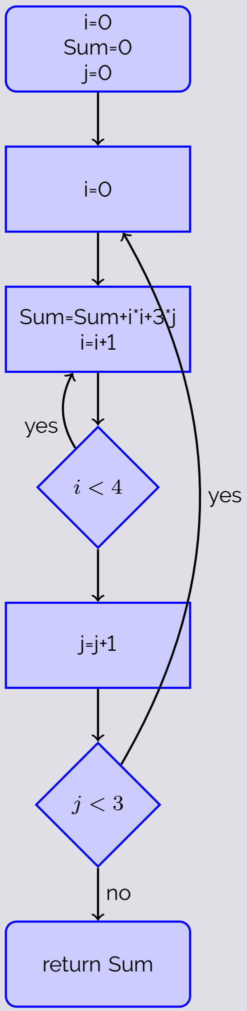
can be expanded as,

$$\sum_{i=0}^{3}\sum_{j=0}^{2}(3j+i^2) = (0^2+3\times 0)+(1^2+3\times 0)+$$
$$(2^2+3\times 0)+(3^2+3\times 0)+$$
$$(0^2+3\times 1)+(1^2+3\times 1)+$$
$$(2^2+3\times 1)+(3^2+3\times 1)+$$
$$(0^2+3\times 2)+(1^2+3\times 2)+$$
$$(2^2+3\times 2)+(3^2+3\times 2)$$
$$=78$$

| $3j+i^2$ | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| j/i | 0 | 1 | 2 | 3 |
| 0 | 0 | 1 | 4 | 9 |
| 1 | 3 | 4 | 7 | 12 |
| 2 | 3 | 4 | 7 | 12 |

## Flowchart - Double Loop

### Example 4 - Double Loop



## Pseudocode – Double Loop

### Example 4 - Double Loop

**Python Pseudocode**

```
# Setting up the initial Sum value
Sum=0

for i in range(0,4):
    for j in range(0,3):
        Sum=Sum+i*i+3*j
print(Sum)
```

The line by line output of the code for Example 3 is:

| Count | i | j | Sum |
|---|---|---|---|
| 0 | | | 0 |
| | | | $Sum = Sum + i^2 + 3*j$ |
| 1 | 0 | 0 | $0+0^2+3*0=0$ |
| 2 | 1 | 0 | $0+1^2+3*0=1$ |
| 3 | 2 | 0 | $1+2^2+3*0=5$ |
| 4 | 3 | 0 | $5+3^2+3*0=14$ |
| 5 | 0 | 1 | $14+0^2+3*1=17$ |
| 6 | 1 | 1 | $17+1^2+3*1=21$ |
| 7 | 2 | 1 | $21+2^2+3*1=28$ |
| 8 | 3 | 1 | $28+3^2+3*1=40$ |
| 9 | 0 | 2 | $40+0^2+3*2=46$ |
| 10 | 1 | 2 | $46+1^2+3*2=53$ |
| 11 | 2 | 2 | $53+2^2+3*2=63$ |
| 12 | 3 | 2 | $63+3^2+3*2=78$ |
| | | | 78 |

# Quick-Find

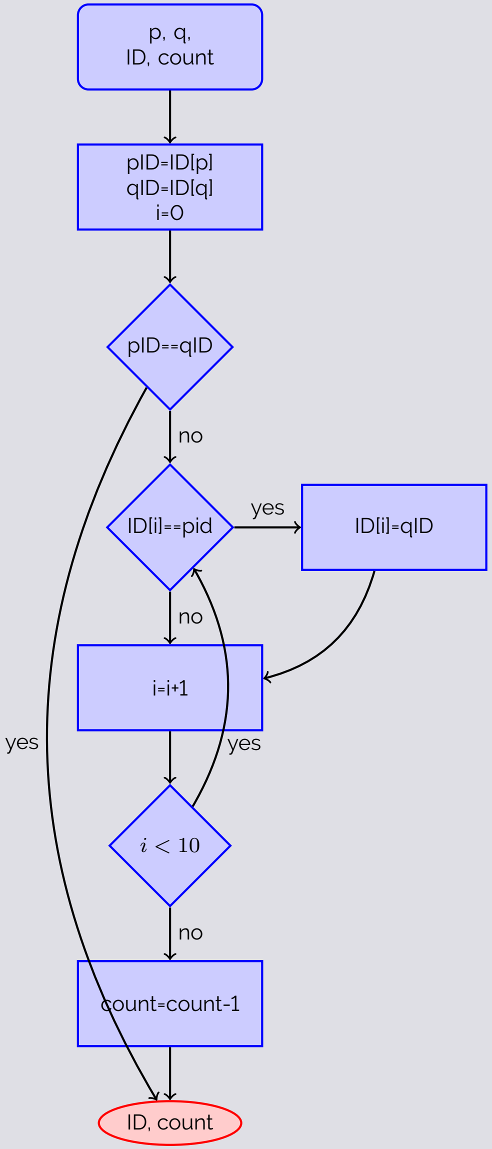
## Pseudocode - Quick Find

### Python Pseudocode

```
def Union(p,q,ID):
        pID=ID[p]
        qID=ID[q]

        if(pID==qID)
                return Connected

         for i in range(0,10):
                if ID[i]==pID:
                        ID[i]=qID
        count=count-1
    return ID
```

## Flowchart

### Quick Find - Connecting



## Pseudocode - Quick Union

### Python Pseudocode Find Root

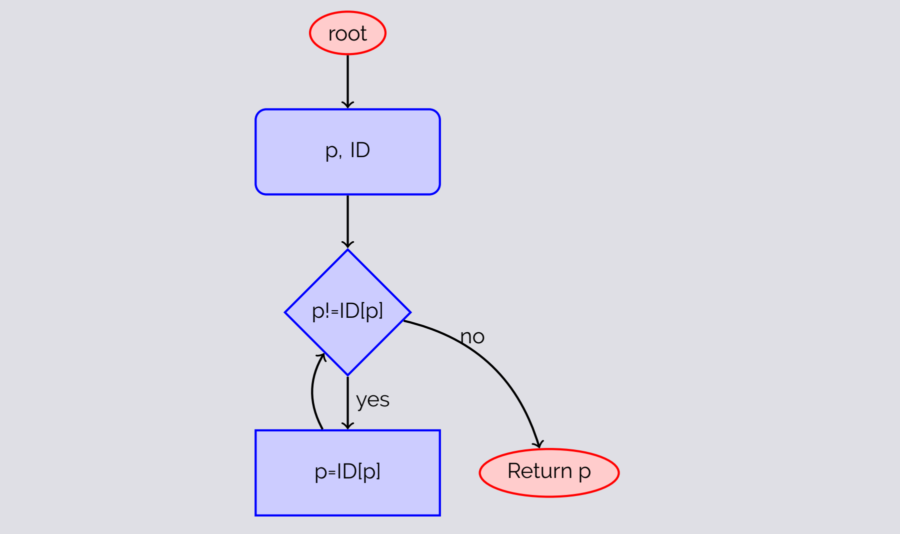
```
def root(p, q, ID):
    while (p!=ID[p]):
            p=ID[p]

    return p
```

### Python Pseudocode Connect

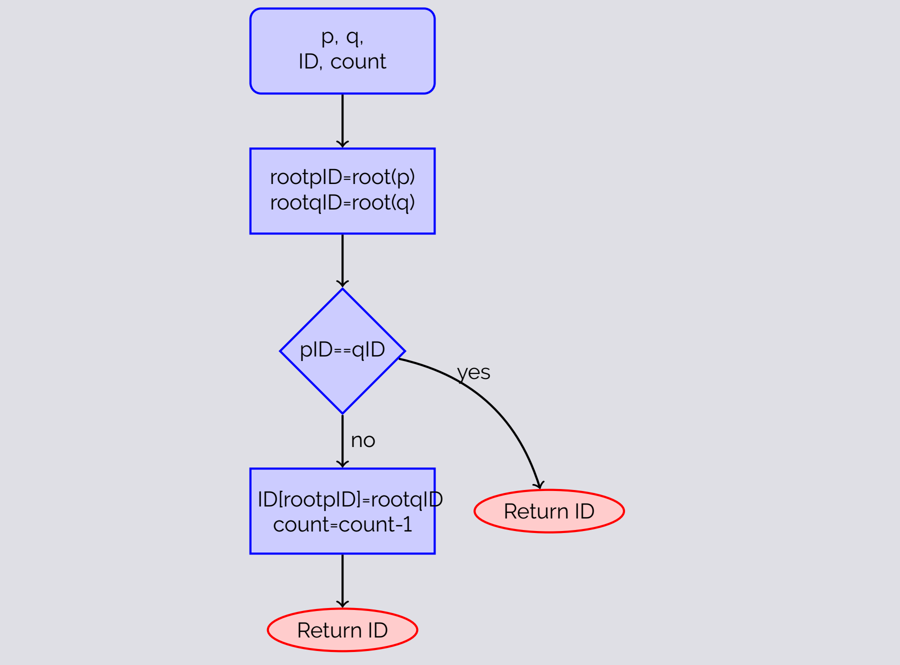
```
def Union(p,q,ID,count):
        root_pID=root(p,ID)
        root_qID=root(q,ID)

        if(root_pID==root_qID):
                ID=ID
        else
            ID[root_pID]=root_qID
            count=count-1
        return ID
```

## Flowchart Quick Union

### Flow Chart Find Root



### Flow Chart Connect Nodes



p, q,
ID, count
rootpID=root(p)
rootqID=root(q)
pID==qID
yes
no
ID[rootpID]=rootqID
count=count-1
Return ID
Return ID