**Homework: Iteration**

1.

a. int i = 0;

while (i < n) {

if (i % i == 0) {

out.print(i);

}

i++;

}

b. int i = 0;

while (i < n) {

if (i % 10 == 0) {

out.print(i);

}

i++;

}

c. int i = 0;

while (Math.pow(2, i) < n) {

out.print(Math.pow(2, i));

i++;

}

2.

a.

|  |  |
| --- | --- |
|  | i = 0  j = 10  n = 0 |
| while (i < j) |  |
| i++ | i = 1 2 3 4 5  j = 10 9 8 7 6  n = 0 1 2 3 4 |
| j-- | i = 1 2 3 4 5  j = 9 8 7 6 5  n = 0 1 2 3 4 |
| n++ | i = 1 2 3 4 5  j = 9 8 7 6 5  n = 1 2 3 4 5 |
|  | **i = 5**  **j = 5**  **n = 5** |

b.

|  |  |
| --- | --- |
|  | i = 0  j = 0  n = 0 |
| while (i < 10) |  |
| i++ | i = 1 2 3 4 5 6 7 8 9 10  j = 0 1 2 3 4 5 6 7 8 9  n = 0 0 1 2 3 4 5 6 7 8 |
| n = n + i – j | i = 1 2 3 4 5 6 7 8 9 10  j = 0 1 2 3 4 5 6 7 8 9  n = 0 1 2 3 4 5 6 7 8 9 |
| j++ | i = 1 2 3 4 5 6 7 8 9 10  j = 1 2 3 4 5 6 7 8 9 10  n = 0 1 2 3 4 5 6 7 8 9 |
|  | **i = 10**  **j = 10**  **n = 9** |

c.

|  |  |
| --- | --- |
|  | i = 10  j = 0  n = 0 |
| while (i > 0) |  |
| i-- | i = 9 8 7 6 5 4 3 2 1 0  j = 0 1 2 3 4 5 6 7 8 9  n = 0 8 14 18 20 20 18 14 8 0 |
| j++ | i = 9 8 7 6 5 4 3 2 1 0  j = 1 2 3 4 5 6 7 8 9 10  n = 0 8 14 18 20 20 18 14 8 0 |
| n = n + i – j | i = 9 8 7 6 5 4 3 2 1 0  j = 1 2 3 4 5 6 7 8 9 10  n = 8 14 18 20 20 18 14 8 0 -10 |
|  | **i = 0**  **j = 10**  **n = -10** |

d.

|  |  |
| --- | --- |
|  | i = 0  j = 10  n = 0 |
| while (i != j) |  |
| i = i + 2 | i = 2 4 6 8  j = 10 8 6 4  n = 0 1 2 3 |
| j = j - 2 | i = 2 4 6 8  j = 8 6 4 2  n = 0 1 2 3 |
| n++ | i = 2 4 6 8  j = 8 6 4 2  n = 1 2 3 4 |
|  | **Infinite loop**  **No final values** |

e.

|  |  |
| --- | --- |
|  | i = 3  j = 4  n = 0 |
| while (i != 0) |  |
| n += j | i = 3 2 1  j = 4 4 4  n = 4 8 12 |
| i-- | i = 2 1 0  j = 4 4 4  n = 4 8 12 |
|  | **i = 0**  **j = 4**  **n = 12** |

3.

int s = 0;

while (i <= 10) {

s = s + i;

i++;

}

4.

int i = 0;

while (i < n) {

v += Math.pow(-1, i) / (2 \* i + 1);

i++;

}

v = v \* 4

5.

while (n < areaBound && m < areaBound) {

sum = Math.pow(n, 2) + Math.pow(m, 2);

}

6.

int i = 0;

while (abs(v – Math.PI()) < epsilon) {

v += Math.pow(-1, i) / (2 \* i + 1);

i++;

}

v = v \* 4