**Problem 8**

1. Rewrite it using no gotos or breaks
   1. j = -3

int i;

for(i = 0; i < 3; i++) {

int t = j + 2;

if( t == 3 ) {

} else if( t == 2) {

j--;

} else if (t == 0) {

j += 2;

} else {

j = 0;

}

if(j <= 0) {

j = 3 – i;

} else {

i = 4;

}

}

1. Rewrite the program using if and goto statements
   1. j = -3

int i;

for(i = 0; i < 3; i++) {

int t = j + 2;

if( t == 3 ) {

} else if( t == 2) {

j--;

} else if (t == 0) {

j += 2;

} else {

j = 0;

}

if(j <= 0) {

j = 3 – i;

} else {

goto nextOut;

}

}

//label

nextOut:

1. Rewrite in java without switch
   1. Int j = -3

for(int i = 0; i < 3; i++) {

int t = j + 2;

if( t == 3 ) {

} else if( t == 2) {

j--;

} else if (t == 0) {

j += 2;

} else {

j = 0;

}

if(j <= 0) {

j = 3 – i;

} else {

break;

}

}

1. Write and explain the operation semantics C program segment

Within the **for** loop, there is the switch statement which matches the value of j + 2 to any of the cases in the block as follow -

If **j + 2 == 3** , then the control goes to case 3, where nothing will happen except for the default

If **j + 2 ==** 2 ,then the **j--** is executed and then breaks from the block

If **j + 2** == 0 ,then the **j += 2** is executed and then breaks from the block.

The **switch** statement is completed, and the **if** block is executed and if **j** is greater than **0**, then we go out of loop. If **j** is negative or 0, then **j = 3-i** is executed and the loop is executed again.