

Semaphores

The aim of this assignment is to make practice with the use of Semaphores in the context of multi-threading programming. Our setting is abstract: the employed Semaphores will not regulate the access to a particular computer device, but only the flux of instructions in the involved concurrent threads.

Question 1

Consider the code for the three threads below. The two Semaphores are named U and W, and we can initially suppose that they both have only 1 permit:

```

// thread 1           // thread 2           // thread 3
while (true)         while (true)         while (true)
{
    W.acquire();      {
                        U.acquire();      {
                        print("AB");      W.acquire();
                        W.release();     print("D");
                    }                   U.release();
                    }                   }
}

```

If only thread 1 is executed, how many times the letters A and B are printed?

Question 2

If only thread 2 is executed, how many times the letter C is printed?

Question 3

Suppose now that the threads 2 and 3 are executed simultaneously. What is the number of D's that will be printed? Is the number of printed C's infinite? Draw the Resource Allocation Graph after two C's and one D are printed, in any order. What can we deduce?

Question 4

Let us suppose now that all threads are executed. Can there be situations where no A's are printed? Answer by giving an example. Same question for the letter C.

Question 5

Can the sequence "DAB" be printed? Same question if both U and W have more than 1 permit.

Question 6

If 3 permits are available for both U and W, can the sequence "ABCCCDCC" be printed?

Question 7

If 3 permits are available for both U and W, can the sequence "CDABABAB" be printed?

Question 8

By using only one semaphore with 3 permits, can you modify the code of the 3 threads so that, once the two letters AB are printed, they will always be followed by a C?

Question 9

Consider the three threads below:

```
// thread 1           // thread 2           // thread 3
x = 1;                x = 2;                z = x + y;
y = 2;                z = 10;
z = 0;
print(z);
```

How many different values of z may be printed?

Question 10

By using only one semaphore with 1 permit, how to change the code of the three threads to make sure that the printed value is always 0?