



Міністерство освіти і науки України
Національний технічний університет України
"Київський політехнічний інститут імені Ігоря Сікорського"
Фізико-технічний інститут

Операційні системи

Лабораторна №8

Виконав:
Студент групи ФБ-82
Козачок Вячеслав
Перевірив:
Кіреєнко О.В.

1 Розминка.

Стандартна задача виробник-споживач. C++

1.1 Input data

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Nulla facilisi nullam vehicula ipsum a arcu cursus vitae. Donec et odio pellentesque diam volutpat commodo. In pellentesque massa placerat dui ultricies lacus sed. Enim nec dui nunc mattis enim. Id consectetur purus ut faucibus pulvinar elementum integer. Pharetra pharetra massa massa ultricies. Pretium quam vulputate dignissim suspendisse in est ante. Sapien nec sagittis aliquam malesuada bibendum. Quis hendrerit dolor magna eget est lorem ipsum. Amet cursus sit amet dictum sit.

Leo integer malesuada nunc vel risus commodo. Arcu non odio euismod lacinia. Orci ac auctor augue mauris. Eu non diam phasellus vestibulum lorem. Nibh tortor id aliquet lectus proin nibh.

Proin fermentum leo vel orci porta non pulvinar neque. Molestie nunc non blandit massa enim nec dui nunc mattis.

Habitant morbi tristique senectus et. Mattis rhoncus urna neque viverra justo nec. Ultricies integer quis auctor

elit sed vulputate mi sit. Justo eget magna fermentum iaculis.

1.2 Code

```

1 #include <stdlib.h>
2 #include <stdio.h>
3 #include <pthread.h>
4 #include <unistd.h>
5 #include <queue>
6 #include <semaphore.h>
7 #include <time.h>
8 using std::queue;
9
10 queue<char*> q;
11 const int max_size = 5;
12 FILE * file;
13 int done = 0;
14
15 sem_t lock;
16 sem_t full;
17 sem_t empty;
18
19 void * produce(void * param)
20 {
21     char * text;
22     while(!done)
23     {
24         sem_wait(&empty);
25         sem_wait(&lock);
26         text = (char*)malloc(255 * sizeof(char));
27         if (fgets(text, 256, file) != NULL)
28         {
29             q.push(text);
30             printf("%s %d [!] Text has been added! \n\n", (char*)param, pthread_self());
31         }
32         else
33         {
34             done = 1;
35         }
36         sem_post(&lock);
37         sem_post(&full);
38         sleep(rand() % 4); // random sleep;

```

```

39     }
40     return NULL;
41 }
42
43 void * consume(void * param)
44 {
45     while(!done || !q.empty()) // when both conditions is False exit the loop;
46     {
47         sem_wait(&full);
48         sem_wait(&lock);
49         if(!q.empty())
50         {
51             printf("%s: %d [!] Text: \n-----\n %s\n", (char*)param, pthread_self(), q.front());
52             q.pop();
53         }
54         sem_post(&lock);
55         sem_post(&empty);
56         sleep(rand() % 4); // random sleep;
57     }
58     return NULL;
59 }
60
61 int main(int argv, char * argc[])
62 {
63     srand(time(0));
64
65     sem_init(&lock, 0, 1);
66     sem_init(&full, 0, 0);
67     sem_init(&empty, 0, max_size);
68     file = fopen("lorem_ipsum.txt", "r");
69
70     pthread_t cons1, cons2;
71     pthread_t prod1, prod2;
72
73     pthread_create(&prod1, NULL, produce, (void*)"producer1");
74     pthread_create(&prod2, NULL, produce, (void*)"producer2");
75     pthread_create(&cons1, NULL, consume, (void*)"consumer1");
76     pthread_create(&cons2, NULL, consume, (void*)"consumer2");
77
78     pthread_join(prod1, NULL);
79     pthread_join(prod2, NULL);
80     pthread_join(cons1, NULL);
81     pthread_join(cons2, NULL);
82
83     return 0;
84 }

```

1.3 Programm output

```

1 producer1 -2053208320 [!] Text has been added!
2
3 consumer2: 2147481344 [!] Text:
4 -----
5 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod
6 -----
7 producer2 -2061601024 [!] Text has been added!
8
9 consumer1: -2069993728 [!] Text:
10 -----
11 tempor incididunt ut labore et dolore magna aliqua. Nulla facilisi nullam vehicula
12 -----
13 producer2 -2061601024 [!] Text has been added!
14
15 producer2 -2061601024 [!] Text has been added!
16
17 producer1 -2053208320 [!] Text has been added!
18
19 producer2 -2061601024 [!] Text has been added!
20
21 producer2 -2061601024 [!] Text has been added!
22
23 consumer2: 2147481344 [!] Text:

```

```

24 -----
25 ipsum a arcu cursus vitae. Donec et odio pellentesque diam volutpat commodo. In
   pellentesque massa
26 -----
27 consumer2: 2147481344 [!] Text:
28 -----
29 placerat dui ultricies lacus sed. Enim nec dui nunc mattis enim. Id consectetur purus
30 -----
31 producer1 -2053208320 [!] Text has been added!
32 -----
33 consumer1: -2069993728 [!] Text:
34 -----
35 ut faucibus pulvinar elementum integer. Pharetra pharetra massa massa ultricies.
36 -----
37 consumer2: 2147481344 [!] Text:
38 -----
39 Pretium quam vulputate dignissim suspendisse in est ante. Sapien nec sagittis
40 -----
41 consumer1: -2069993728 [!] Text:
42 -----
43 aliquam malesuada bibendum. Quis hendrerit dolor magna eget est lorem ipsum. Amet cursus
   sit amet dictum sit.
44 -----
45 producer2 -2061601024 [!] Text has been added!
46 -----
47 consumer1: -2069993728 [!] Text:
48 -----
49 Leo integer malesuada nunc vel risus commodo. Arcu non odio euismod lacinia.
50 -----
51 consumer1: -2069993728 [!] Text:
52 -----
53 Orci ac auctor augue mauris. Eu non diam phasellus vestibulum lorem. Nibh tortor id aliquet
   lectus proin nibh.
54 -----
55 producer2 -2061601024 [!] Text has been added!
56 -----
57 consumer1: -2069993728 [!] Text:
58 -----
59 Proin fermentum leo vel orci porta non pulvinar neque. Molestie nunc non blandit massa enim
   nec dui nunc mattis.
60 -----
61 producer1 -2053208320 [!] Text has been added!
62 -----
63 consumer2: 2147481344 [!] Text:
64 -----
65 Habitant morbi tristique senectus et. Mattis rhoncus urna neque viverra justo nec.
   Ultricies integer quis auctor
66 -----
67 producer1 -2053208320 [!] Text has been added!
68 -----
69 consumer1: -2069993728 [!] Text:
70 -----
71 elit sed vulputate mi sit. Justo eget magna fermentum iaculis.
72 -----

```

2 Продовження розминки.

Теж саме, але не на семафорах, а на м'ютексі і умовних змінних. C++

2.1 Input file

Такий самий (Див 1.1)

2.2 Code

```

1 #include <stdlib.h>
2 #include <stdio.h>
3 #include <pthread.h>
4 #include <unistd.h>
5 #include <queue>
6 #include <semaphore.h>
7 #include <time.h>
8 using std::queue;
9
10 queue<char*> q;
11 FILE * file;
12 int done = 0;
13
14 pthread_mutex_t mutex;
15
16 void * produce(void * param)
17 {
18     char * text;
19     while(!done)
20     {
21         pthread_mutex_lock(&mutex);
22         text = (char*)malloc(255 * sizeof(char));
23         if (fgets(text, 256, file) != NULL)
24         {
25             q.push(text);
26             printf("%s %d [!] Text has been added! \n\n", (char*)param, pthread_self());
27         }
28         else
29         {
30             done = 1;
31         }
32         pthread_mutex_unlock(&mutex);
33         sleep(rand() % 4); // random sleep;
34     }
35     return NULL;
36 }
37
38 void * consume(void * param)
39 {
40     while(!done || !q.empty()) // when both conditions is False exit the loop;
41     {
42         pthread_mutex_lock(&mutex);
43         if(!q.empty())
44         {
45             printf("%s: %d [!] Text: \n-----\n %s-----\n", (char
46 *)param , pthread_self(), q.front());
47             q.pop();
48         }
49         pthread_mutex_unlock(&mutex);
50         sleep(rand() % 4); // random sleep;
51     }
52     return NULL;
53 }
54
55 int main(int argv, char * argc[])
56 {
57     srand(time(0));
58
59     file = fopen("lorem_ipsum.txt", "r");

```

```

60
61     pthread_t cons1, cons2;
62     pthread_t prod1, prod2;
63     pthread_mutex_init(&mutex, NULL);
64
65     pthread_create(&prod1, NULL, produce, (void*)"producer1");
66     pthread_create(&prod2, NULL, produce, (void*)"producer2");
67     pthread_create(&cons1, NULL, consume, (void*)"consumer1");
68     pthread_create(&cons2, NULL, consume, (void*)"consumer2");
69
70     pthread_join(prod1, NULL);
71     pthread_join(prod2, NULL);
72     pthread_join(cons1, NULL);
73     pthread_join(cons2, NULL);
74
75     return 0;
76 }

```

2.3 Program output

```

1 producer1 -1394247936 [!] Text has been added!
2
3 producer2 -1402640640 [!] Text has been added!
4
5 producer2 -1402640640 [!] Text has been added!
6
7 consumer1: -1411033344 [!] Text:
8 -----
9 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod
10 -----
11 consumer2: -1419426048 [!] Text:
12 -----
13 tempor incididunt ut labore et dolore magna aliqua. Nulla facilisi nullam vehicula
14 -----
15 consumer1: -1411033344 [!] Text:
16 -----
17 ipsum a arcu cursus vitae. Donec et odio pellentesque diam volutpat commodo. In
18 pellentesque massa
19 -----
20 producer1 -1394247936 [!] Text has been added!
21
22 producer2 -1402640640 [!] Text has been added!
23
24 consumer1: -1411033344 [!] Text:
25 -----
26 placerat dui ultricies lacus sed. Enim nec dui nunc mattis enim. Id consectetur purus
27 -----
28 consumer2: -1419426048 [!] Text:
29 -----
30 ut faucibus pulvinar elementum integer. Pharetra pharetra massa massa ultricies.
31 -----
32 producer1 -1394247936 [!] Text has been added!
33
34 producer2 -1402640640 [!] Text has been added!
35
36 consumer1: -1411033344 [!] Text:
37 -----
38 Pretium quam vulputate dignissim suspendisse in est ante. Sapien nec sagittis
39 -----
40 consumer1: -1411033344 [!] Text:
41 -----
42 aliquam malesuada bibendum. Quis hendrerit dolor magna eget est lorem ipsum. Amet cursus
43 sit amet dictum sit.
44 -----
45 producer1 -1394247936 [!] Text has been added!
46
47 producer2 -1402640640 [!] Text has been added!
48
49 consumer1: -1411033344 [!] Text:
50 -----
51 Leo integer malesuada nunc vel risus commodo. Arcu non odio euismod lacinia.
52 -----
53 consumer1: -1411033344 [!] Text:

```

```
52 -----
53 Orci ac auctor augue mauris. Eu non diam phasellus vestibulum lorem. Nibh tortor id aliquet
    lectus proin nibh.
54 -----
55 producer2 -1402640640 [!] Text has been added!
56
57 consumer1: -1411033344 [!] Text:
58 -----
59 Proin fermentum leo vel orci porta non pulvinar neque. Molestie nunc non blandit massa enim
    nec dui nunc mattis.
60 -----
61 producer1 -1394247936 [!] Text has been added!
62
63 consumer2: -1419426048 [!] Text:
64 -----
65 Habitant morbi tristique senectus et. Mattis rhoncus urna neque viverra justo nec.
    Ultricies integer quis auctor
66 -----
67 producer1 -1394247936 [!] Text has been added!
68
69 consumer2: -1419426048 [!] Text:
70 -----
71 elit sed vulputate mi sit. Justo eget magna fermentum iaculis.
```

3 Продовження розминки для тих, хто шукає пригод. Взаємне блокування

Модифікуйте програму п. 1 так, щоби викликати взаємне блокування. Для цього поміняйте місцями семафори. Переконайтесь у факті взаємного блокування і отримайте задоволення. C++

3.1 Input file

Такий самий (Див 1.1)

3.2 Code

```

1 #include <stdlib.h>
2 #include <stdio.h>
3 #include <pthread.h>
4 #include <unistd.h>
5 #include <queue>
6 #include <semaphore.h>
7 #include <time.h>
8 using std::queue;
9
10 queue<char*> q;
11 const int max_size = 5;
12 FILE * file;
13 int done = 0;
14
15 sem_t lock;
16 sem_t full;
17 sem_t empty;
18
19 void * produce(void * param)
20 {
21     char * text;
22     while(!done)
23     {
24         sem_wait(&lock);
25         sem_wait(&empty);
26         text = (char*)malloc(255 * sizeof(char));
27         if (fgets(text, 256, file) != NULL)
28         {
29             q.push(text);
30             printf("%s %d [!] Text has been added! \n\n", (char*)param, pthread_self());
31         }
32         else
33         {
34             done = 1;
35         }
36         sem_post(&lock);
37         sem_post(&full);
38         sleep(rand() % 4); // random sleep;
39     }
40     return NULL;
41 }
42
43 void * consume(void * param)
44 {
45     while(!done || !q.empty()) // when both conditions is False exit the loop;
46     {
47         sem_wait(&lock);
48         sem_wait(&full);
49         if(!q.empty())
50         {
51             printf("%s: %d [!] Text: \n-----\n %s\n", (char*)param, pthread_self(), q.front());
52             q.pop();
53         }
54         sem_post(&lock);
55         sem_post(&empty);
56     }

```



```

57     sleep(rand() % 4); // random sleep;
58 }
59 return NULL;
60 }
61
62 int main(int argv, char * argc[])
63 {
64     srand(time(0));
65
66     sem_init(&lock, 0, 1);
67     sem_init(&full, 0, 0);
68     sem_init(&empty, 0, max_size);
69     file = fopen("lorem_ipsum.txt", "r");
70
71     pthread_t cons1, cons2;
72     pthread_t prod1, prod2;
73
74     pthread_create(&prod1, NULL, produce, (void*)"producer1");
75     pthread_create(&prod2, NULL, produce, (void*)"producer2");
76     pthread_create(&cons1, NULL, consume, (void*)"consumer1");
77     pthread_create(&cons2, NULL, consume, (void*)"consumer2");
78
79     pthread_join(prod1, NULL);
80     pthread_join(prod2, NULL);
81     pthread_join(cons1, NULL);
82     pthread_join(cons2, NULL);
83
84     return 0;
85 }

```

3.3 Program output

```

1 producer1 457340672 [!] Text has been added!
2
3 producer2 448947968 [!] Text has been added!
4
5 consumer1: 335542016 [!] Text:
6 -----
7 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod
8 -----
9 consumer2: 440555264 [!] Text:
10 -----
11 tempor incididunt ut labore et dolore magna aliqua. Nulla facilisi nullam vehicula
12 -----
13
14 // Programm stucks

```

4 Індивідуальне завдання. Варіант 1.

4.1 Code

```

1 #include <iostream>
2 #include <pthread.h>
3 #include <queue>
4 #include <semaphore.h>
5 #include <time.h>
6 #include <unistd.h>
7
8 struct thread_info
9 {
10     int thread_number;
11     char * name;
12     pthread_t * thread;
13 };
14
15 std::queue<thread_info*> on_board;
16 std::queue<thread_info*> to_cancel;
17
18 sem_t busy_places, boarding, disembarkation;
19
20 int paratroopers_on_board = 0;

```

```

21 int marines_on_board = 0;
22 int max_places;
23 int handled_soldiers = 0;
24
25 void trip()
26 {
27     printf("\n");
28     for(int i = 0; i < 3; i++)
29     {
30         printf("%16s\n", "Ferry in trip!");
31         sleep(1);
32     }
33     printf("\n");
34
35     // Allow troops leave the ferry
36     sem_post(&disembarkation);
37 }
38
39 void get_on_board(thread_info * info)
40 {
41     sem_wait(&boarding);
42     if(info->name == "Marine")
43     {
44         if((marines_on_board == max_places/2 && paratroopers_on_board != 0) || (
45         paratroopers_on_board > max_places/2))
46         {
47             printf("%12s %d canceled ", info->name, info->thread_number);
48             to_cancel.push(info);
49         }
50         else
51         {
52             printf("%12s %d boarded ", info->name, info->thread_number);
53             on_board.push(info);
54             marines_on_board++;
55         }
56     }
57     else if (info->name == "Paratrooper")
58     {
59         if((paratroopers_on_board == max_places/2 && marines_on_board != 0) || (
60         marines_on_board > max_places/2))
61         {
62             printf("%12s %d canceled ", info->name, info->thread_number);
63             to_cancel.push(info);
64         }
65         else
66         {
67             printf("%12s %d boarded ", info->name, info->thread_number);
68             on_board.push(info);
69             paratroopers_on_board++;
70         }
71     }
72
73     printf("< Current Time: %d \n", time(0));
74     handled_soldiers++;
75     sem_post(&boarding);
76 }
77
78 void get_out_of_board(thread_info * info)
79 {
80     sem_wait(&disembarkation);
81     printf("%12s %d disembarked. < Current time: %d\n", info->name, info->thread_number,
82     time(0));
83     sleep(1);
84     sem_post(&disembarkation);
85 }
86
87 void clear_ferry()
88 {
89     while(on_board.size() != 0)
90     {
91         // printf("Waiting clear\n");
92         pthread_join(*(on_board.front()->thread), NULL);
93         on_board.pop();
94     }
95 }

```

```

91     }
92 }
93
94 void cancel_threads()
95 {
96     while(to_cancel.size() != 0)
97     {
98         pthread_cancel(*(to_cancel.front()->thread));
99         to_cancel.pop();
100     }
101 }
102
103
104 void * thread_work(void * param)
105 {
106     thread_info * info = (thread_info*)param;
107     sleep(rand() % 3 + 1);
108
109     get_on_board(info);
110     get_out_of_board(info);
111     pthread_exit(NULL);
112 }
113
114 int main(int argv, char * argc[])
115 {
116     std::cout << "Enter N: ";
117     int n;
118     std::cin >> n;
119     if(std::cin.fail())
120     {
121         std::cout << "Exiting ...\n";
122         return 1;
123     }
124
125     srand(time(0));
126     max_places = n * 2;
127     printf("\n%16s\n", "INFORMATION");
128     printf("%16s: %d\n", "Available places", max_places);
129     printf("%16s: %d\n", "Paratroopers", max_places);
130     printf("%16s: %d\n", "Marins", max_places);
131     printf("%20s\n", "-----");
132
133     // Semaphores
134     sem_init(&boarding, 0, 1);
135     sem_init(&disembarkation, 0, 0);
136
137     // Soldiers
138     pthread_t * marine = new pthread_t[max_places];
139     pthread_t * paratrooper = new pthread_t[max_places];
140
141     int counter = 1;
142     thread_info * info;
143     for(int i = 0; i < max_places; i++)
144     {
145         // Creating marine
146         info = new thread_info;
147         info->thread_number = counter;
148         info->thread = &marine[i];
149         info->name = "Marine";
150         pthread_create(&marine[i], NULL, thread_work, (void*)info);
151
152         // Creating paratrooper
153         info = new thread_info;
154         info->thread_number = counter;
155         info->thread = &paratrooper[i];
156         info->name = "Paratrooper";
157         pthread_create(&paratrooper[i], NULL, thread_work, ((void*)info));
158         counter++;
159     }
160
161     // wait till all soldiers is canceled or boarded
162     while(handled_soldiers != max_places*2)
163     {

```

```
164     sleep(1);
165     continue;
166 }
167
168 // remove soldiers that could not board ferry
169 printf("\nCleaning all threads troops that are not in ferry\n");
170 cancel_threads();
171
172 trip();
173
174 // clear ferry on another side
175 printf("Eject all troops on coast\n\n");
176 clear_ferry();
177
178 return 0;
179 }
```

4.2 Program Output

$N = 2$

```

1 Enter N: 2
2
3      INFORMATION
4 Available places: 4
5     Paratroopers: 4
6       Marines: 4
7     -----
8 Paratrooper 0 boarded < Current Time: 1588159550
9 Paratrooper 2 boarded < Current Time: 1588159551
10    Marine 0 boarded < Current Time: 1588159552
11 Paratrooper 1 canceled < Current Time: 1588159553
12    Marine 2 boarded < Current Time: 1588159554
13    Marine 3 canceled < Current Time: 1588159555
14 Paratrooper 3 canceled < Current Time: 1588159556
15    Marine 1 canceled < Current Time: 1588159557
16
17 Cleaning all threads troops that are not in ferry
18
19 Ferry in trip!
20 Ferry in trip!
21 Ferry in trip!
22
23 Eject all troops on coast
24
25 Paratrooper 0 disembarked. < Current time: 1588159561
26 Paratrooper 2 disembarked. < Current time: 1588159562
27    Marine 0 disembarked. < Current time: 1588159563
28    Marine 2 disembarked. < Current time: 1588159564

```

N = 3

```

1 Enter N: 3
2
3      INFORMATION
4 Available places: 6
5     Paratroopers: 6
6         Marins: 6
7         -----
8         Marine 2 boarded < Current Time: 1588247530
9         Marine 4 boarded < Current Time: 1588247530
10        Paratrooper 4 boarded < Current Time: 1588247530
11        Marine 5 boarded < Current Time: 1588247530
12        Paratrooper 6 boarded < Current Time: 1588247530
13        Paratrooper 1 boarded < Current Time: 1588247531
14        Marine 3 canceled < Current Time: 1588247531
15        Paratrooper 3 canceled < Current Time: 1588247531
16        Marine 6 canceled < Current Time: 1588247531
17        Marine 1 canceled < Current Time: 1588247532
18        Paratrooper 2 canceled < Current Time: 1588247532
19        Paratrooper 5 canceled < Current Time: 1588247532
20
21 Cleaning all threads troops that are not in ferry
22
23 Ferry in trip!
24 Ferry in trip!
25 Ferry in trip!
26
27 Eject all troops on coast
28
29         Marine 2 disembarked. < Current time: 1588247535
30         Marine 4 disembarked. < Current time: 1588247536
31        Paratrooper 4 disembarked. < Current time: 1588247537
32         Marine 5 disembarked. < Current time: 1588247538
33        Paratrooper 6 disembarked. < Current time: 1588247539
34        Paratrooper 1 disembarked. < Current time: 1588247540

```

```

1 Enter N: 3
2
3      INFORMATION
4 Available places: 6
5     Paratroopers: 6
6         Marins: 6
7         -----
8         Marine 3 boarded < Current Time: 1588247916
9         Marine 4 boarded < Current Time: 1588247916
10        Marine 1 boarded < Current Time: 1588247917
11        Marine 2 boarded < Current Time: 1588247917
12        Paratrooper 3 canceled < Current Time: 1588247917
13        Marine 5 boarded < Current Time: 1588247917
14        Paratrooper 4 canceled < Current Time: 1588247918
15        Paratrooper 5 canceled < Current Time: 1588247918
16        Marine 6 boarded < Current Time: 1588247918
17        Paratrooper 1 canceled < Current Time: 1588247919
18        Paratrooper 2 canceled < Current Time: 1588247919
19        Paratrooper 6 canceled < Current Time: 1588247919
20
21 Cleaning all threads troops that are not in ferry
22
23 Ferry in trip!
24 Ferry in trip!
25 Ferry in trip!
26
27 Eject all troops on coast
28
29         Marine 3 disembarked. < Current time: 1588247922
30         Marine 4 disembarked. < Current time: 1588247923
31         Marine 1 disembarked. < Current time: 1588247924
32         Marine 2 disembarked. < Current time: 1588247925
33         Marine 5 disembarked. < Current time: 1588247926
34         Marine 6 disembarked. < Current time: 1588247927

```

```

1 Enter N: 3

```

```

2
3      INFORMATION
4 Available places: 6
5     Paratroopers: 6
6       Marins: 6
7       -----
8 Paratrooper 1 boarded < Current Time: 1588248044
9 Paratrooper 2 boarded < Current Time: 1588248044
10 Paratrooper 3 boarded < Current Time: 1588248044
11 Paratrooper 4 boarded < Current Time: 1588248044
12 Paratrooper 5 boarded < Current Time: 1588248044
13     Marine 2 canceled < Current Time: 1588248046
14     Marine 3 canceled < Current Time: 1588248046
15     Marine 5 canceled < Current Time: 1588248046
16     Marine 6 canceled < Current Time: 1588248046
17 Paratrooper 6 boarded < Current Time: 1588248046
18     Marine 1 canceled < Current Time: 1588248047
19     Marine 4 canceled < Current Time: 1588248047
20
21 Cleaning all threads troops that are not in ferry
22
23   Ferry in trip!
24   Ferry in trip!
25   Ferry in trip!
26
27 Eject all troops on coast
28
29 Paratrooper 1 disembarked. < Current time: 1588248050
30 Paratrooper 2 disembarked. < Current time: 1588248051
31 Paratrooper 3 disembarked. < Current time: 1588248052
32 Paratrooper 4 disembarked. < Current time: 1588248053
33 Paratrooper 5 disembarked. < Current time: 1588248054
34 Paratrooper 6 disembarked. < Current time: 1588248055

```

N = 5

```

1 Enter N: 5
2
3      INFORMATION
4 Available places: 10
5     Paratroopers: 10
6       Marins: 10
7     -----
8       Marine 1 boarded < Current Time: 1588159802
9 Paratrooper 3 boarded < Current Time: 1588159803
10      Marine 4 boarded < Current Time: 1588159804
11 Paratrooper 4 boarded < Current Time: 1588159805
12 Paratrooper 7 boarded < Current Time: 1588159806
13      Marine 9 boarded < Current Time: 1588159807
14      Marine 0 boarded < Current Time: 1588159808
15 Paratrooper 0 boarded < Current Time: 1588159809
16 Paratrooper 2 boarded < Current Time: 1588159810
17      Marine 3 boarded < Current Time: 1588159811
18      Marine 2 canceled < Current Time: 1588159812
19      Marine 5 canceled < Current Time: 1588159813
20 Paratrooper 5 canceled < Current Time: 1588159814
21      Marine 7 canceled < Current Time: 1588159815
22 Paratrooper 8 canceled < Current Time: 1588159816
23 Paratrooper 1 canceled < Current Time: 1588159817
24      Marine 6 canceled < Current Time: 1588159818
25 Paratrooper 6 canceled < Current Time: 1588159819
26      Marine 8 canceled < Current Time: 1588159820
27 Paratrooper 9 canceled < Current Time: 1588159821
28
29 Cleaning all threads troops that are not in ferry
30
31 Ferry in trip!
32 Ferry in trip!
33 Ferry in trip!
34
35 Eject all troops on coast
36
37      Marine 1 disembarked. < Current time: 1588159825
38 Paratrooper 3 disembarked. < Current time: 1588159826
39      Marine 4 disembarked. < Current time: 1588159827
40 Paratrooper 4 disembarked. < Current time: 1588159828
41 Paratrooper 7 disembarked. < Current time: 1588159829
42      Marine 9 disembarked. < Current time: 1588159830
43      Marine 0 disembarked. < Current time: 1588159831
44 Paratrooper 0 disembarked. < Current time: 1588159832
45 Paratrooper 2 disembarked. < Current time: 1588159833
46      Marine 3 disembarked. < Current time: 1588159834

```


N = 7

```

1 Enter N: 7
2
3 INFORMATION
4 Available places: 14
5 Paratroopers: 14
6 Marins: 14
7 -----
8 Marine 4 boarded < Current Time: 1588247664
9 Paratrooper 5 boarded < Current Time: 1588247664
10 Paratrooper 6 boarded < Current Time: 1588247664
11 Paratrooper 7 boarded < Current Time: 1588247664
12 Marine 8 boarded < Current Time: 1588247664
13 Paratrooper 8 boarded < Current Time: 1588247664
14 Paratrooper 9 boarded < Current Time: 1588247664
15 Paratrooper 12 boarded < Current Time: 1588247664
16 Marine 13 boarded < Current Time: 1588247664
17 Marine 14 boarded < Current Time: 1588247664
18 Paratrooper 14 boarded < Current Time: 1588247664
19 Marine 1 boarded < Current Time: 1588247665
20 Paratrooper 2 canceled < Current Time: 1588247665
21 Marine 3 boarded < Current Time: 1588247665
22 Paratrooper 4 canceled < Current Time: 1588247665
23 Marine 5 boarded < Current Time: 1588247665
24 Marine 7 canceled < Current Time: 1588247665
25 Paratrooper 10 canceled < Current Time: 1588247665
26 Marine 11 canceled < Current Time: 1588247665
27 Paratrooper 11 canceled < Current Time: 1588247665
28 Paratrooper 1 canceled < Current Time: 1588247666
29 Marine 2 canceled < Current Time: 1588247666
30 Paratrooper 3 canceled < Current Time: 1588247666
31 Marine 6 canceled < Current Time: 1588247666
32 Marine 9 canceled < Current Time: 1588247666
33 Marine 10 canceled < Current Time: 1588247666
34 Marine 12 canceled < Current Time: 1588247666
35 Paratrooper 13 canceled < Current Time: 1588247666
36
37 Cleaning all threads troops that are not in ferry
38
39 Ferry in trip!
40 Ferry in trip!
41 Ferry in trip!
42
43 Eject all troops on coast
44
45 Marine 4 disembarked. < Current time: 1588247669
46 Paratrooper 5 disembarked. < Current time: 1588247670
47 Paratrooper 6 disembarked. < Current time: 1588247671
48 Paratrooper 7 disembarked. < Current time: 1588247672
49 Marine 8 disembarked. < Current time: 1588247673
50 Paratrooper 8 disembarked. < Current time: 1588247674
51 Paratrooper 9 disembarked. < Current time: 1588247675
52 Paratrooper 12 disembarked. < Current time: 1588247676
53 Marine 13 disembarked. < Current time: 1588247677
54 Marine 14 disembarked. < Current time: 1588247678
55 Paratrooper 14 disembarked. < Current time: 1588247679
56 Marine 1 disembarked. < Current time: 1588247680
57 Marine 3 disembarked. < Current time: 1588247681
58 Marine 5 disembarked. < Current time: 1588247682

```

N = 20

```

1 Enter N: 20
2
3 INFORMATION
4 Available places: 40
5 Paratroopers: 40
6 Marins: 40
7 -----
8 Marine 1 boarded < Current Time: 1588247722
9 Marine 2 boarded < Current Time: 1588247722
10 Paratrooper 2 boarded < Current Time: 1588247722
11 Marine 3 boarded < Current Time: 1588247722
12 Paratrooper 4 boarded < Current Time: 1588247722
13 Marine 5 boarded < Current Time: 1588247722
14 Marine 6 boarded < Current Time: 1588247722
15 Marine 7 boarded < Current Time: 1588247722
16 Marine 8 boarded < Current Time: 1588247722
17 Marine 9 boarded < Current Time: 1588247722
18 Marine 10 boarded < Current Time: 1588247722
19 Paratrooper 11 boarded < Current Time: 1588247722
20 Paratrooper 13 boarded < Current Time: 1588247722
21 Marine 15 boarded < Current Time: 1588247722
22 Marine 16 boarded < Current Time: 1588247722
23 Paratrooper 16 boarded < Current Time: 1588247722
24 Paratrooper 17 boarded < Current Time: 1588247722
25 Marine 18 boarded < Current Time: 1588247722
26 Marine 19 boarded < Current Time: 1588247722
27 Marine 20 boarded < Current Time: 1588247722
28 Marine 21 boarded < Current Time: 1588247722
29 Marine 24 boarded < Current Time: 1588247722
30 Paratrooper 24 boarded < Current Time: 1588247722
31 Marine 26 boarded < Current Time: 1588247722
32 Paratrooper 26 boarded < Current Time: 1588247722
33 Paratrooper 27 boarded < Current Time: 1588247722
34 Paratrooper 29 boarded < Current Time: 1588247722
35 Marine 30 boarded < Current Time: 1588247722
36 Marine 31 boarded < Current Time: 1588247722
37 Paratrooper 31 boarded < Current Time: 1588247722
38 Marine 34 boarded < Current Time: 1588247722
39 Paratrooper 34 boarded < Current Time: 1588247722
40 Paratrooper 35 boarded < Current Time: 1588247722
41 Paratrooper 36 boarded < Current Time: 1588247722
42 Paratrooper 37 boarded < Current Time: 1588247722
43 Marine 38 canceled < Current Time: 1588247722
44 Paratrooper 39 boarded < Current Time: 1588247722
45 Paratrooper 40 boarded < Current Time: 1588247722
46 Marine 4 canceled < Current Time: 1588247723
47 Paratrooper 5 boarded < Current Time: 1588247723
48 Paratrooper 7 boarded < Current Time: 1588247723
49 Paratrooper 8 boarded < Current Time: 1588247723
50 Paratrooper 12 canceled < Current Time: 1588247723
51 Marine 14 canceled < Current Time: 1588247723
52 Paratrooper 14 canceled < Current Time: 1588247723
53 Paratrooper 20 canceled < Current Time: 1588247723
54 Paratrooper 21 canceled < Current Time: 1588247723
55 Marine 22 canceled < Current Time: 1588247723
56 Paratrooper 23 canceled < Current Time: 1588247723
57 Marine 25 canceled < Current Time: 1588247723
58 Paratrooper 25 canceled < Current Time: 1588247723
59 Marine 32 canceled < Current Time: 1588247723
60 Marine 33 canceled < Current Time: 1588247723
61 Paratrooper 33 canceled < Current Time: 1588247723
62 Marine 35 canceled < Current Time: 1588247723
63 Marine 36 canceled < Current Time: 1588247723
64 Paratrooper 1 canceled < Current Time: 1588247724
65 Paratrooper 3 canceled < Current Time: 1588247724
66 Paratrooper 6 canceled < Current Time: 1588247724
67 Paratrooper 9 canceled < Current Time: 1588247724
68 Paratrooper 10 canceled < Current Time: 1588247724
69 Marine 11 canceled < Current Time: 1588247724
70 Marine 12 canceled < Current Time: 1588247724
71 Marine 13 canceled < Current Time: 1588247724

```

```

72 Paratrooper 15 canceled < Current Time: 1588247724
73     Marine 17 canceled < Current Time: 1588247724
74 Paratrooper 18 canceled < Current Time: 1588247724
75 Paratrooper 19 canceled < Current Time: 1588247724
76 Paratrooper 22 canceled < Current Time: 1588247724
77     Marine 23 canceled < Current Time: 1588247724
78     Marine 27 canceled < Current Time: 1588247724
79     Marine 28 canceled < Current Time: 1588247724
80 Paratrooper 28 canceled < Current Time: 1588247724
81     Marine 29 canceled < Current Time: 1588247724
82 Paratrooper 30 canceled < Current Time: 1588247724
83 Paratrooper 32 canceled < Current Time: 1588247724
84     Marine 37 canceled < Current Time: 1588247724
85 Paratrooper 38 canceled < Current Time: 1588247724
86     Marine 39 canceled < Current Time: 1588247724
87     Marine 40 canceled < Current Time: 1588247724
88
89 Cleaning all threads troops that are not in ferry
90
91     Ferry in trip!
92     Ferry in trip!
93     Ferry in trip!
94
95 Eject all troops on coast
96
97     Marine 1 disembarked. < Current time: 1588247727
98     Marine 2 disembarked. < Current time: 1588247728
99 Paratrooper 2 disembarked. < Current time: 1588247729
100     Marine 3 disembarked. < Current time: 1588247730
101 Paratrooper 4 disembarked. < Current time: 1588247731
102     Marine 5 disembarked. < Current time: 1588247732
103     Marine 6 disembarked. < Current time: 1588247733
104     Marine 7 disembarked. < Current time: 1588247734
105     Marine 8 disembarked. < Current time: 1588247735
106     Marine 9 disembarked. < Current time: 1588247736
107     Marine 10 disembarked. < Current time: 1588247737
108 Paratrooper 11 disembarked. < Current time: 1588247738
109 Paratrooper 13 disembarked. < Current time: 1588247739
110     Marine 15 disembarked. < Current time: 1588247740
111     Marine 16 disembarked. < Current time: 1588247741
112 Paratrooper 16 disembarked. < Current time: 1588247742
113 Paratrooper 17 disembarked. < Current time: 1588247743
114     Marine 18 disembarked. < Current time: 1588247744
115     Marine 19 disembarked. < Current time: 1588247745
116     Marine 20 disembarked. < Current time: 1588247746
117     Marine 21 disembarked. < Current time: 1588247747
118     Marine 24 disembarked. < Current time: 1588247748
119 Paratrooper 24 disembarked. < Current time: 1588247749
120     Marine 26 disembarked. < Current time: 1588247750
121 Paratrooper 26 disembarked. < Current time: 1588247751
122 Paratrooper 27 disembarked. < Current time: 1588247752
123 Paratrooper 29 disembarked. < Current time: 1588247753
124     Marine 30 disembarked. < Current time: 1588247754
125     Marine 31 disembarked. < Current time: 1588247755
126 Paratrooper 31 disembarked. < Current time: 1588247756
127     Marine 34 disembarked. < Current time: 1588247757
128 Paratrooper 34 disembarked. < Current time: 1588247758
129 Paratrooper 35 disembarked. < Current time: 1588247759
130 Paratrooper 36 disembarked. < Current time: 1588247760
131 Paratrooper 37 disembarked. < Current time: 1588247761
132 Paratrooper 39 disembarked. < Current time: 1588247762
133 Paratrooper 40 disembarked. < Current time: 1588247763
134 Paratrooper 5 disembarked. < Current time: 1588247764
135 Paratrooper 7 disembarked. < Current time: 1588247765
136 Paratrooper 8 disembarked. < Current time: 1588247766

```

5 Висновки

Впродовж виконання цієї лабораторної роботи я дізнався: як створювати багатопотокові програми, блокувати потоки, коли це потрібно за допомогою семафорів, мютексів. Розглянув такий варіант семафору як `locker`. Передавати в потоки дані. Найбільш зручно використовувати структури для передачі даних у потоки.