

# Raport Projektu

Temat 16 – Stacja Narciarska

Piotr Dudczyk 151419

## Najważniejsze informacje potrzebne do uruchomienia:

1. Program rozpoczynamy z terminala komendą **./Main** który należy poprzedzić uruchomieniem skryptu **./build.sh** odpowiedzialnym za zbudowania wszystkich niezbędnych plików
2. w main'ie znajdują się parametry które możemy zmienić tzn. Tp – czas otwarcia stacji, Tk- czas zamknięcia stacji(zmianę należy uwzględnić również w Kasjer.c), scale\_duration-czas trwania symulacji
3. dodatkowo w procesie narciarz mamy parametr N-określający max. Liczbę osób na peronie w danym momencie
4. większość funkcji została rozdzielona do odpowiednich plików: **pamiec\_dzielona.h**, **pamiec\_dzielona\_2.h**, **queue.h**, **semafory.h**
5. wszystkie wykorzystywane struktury zapisane są w **structs.h**
6. w kodzie występuje obsługa sygnałów które powodują zatrzymanie krzesełek w razie awarii żeby wywołać działanie sygnału należy wywołać w terminalu **komendę kill -SIGUSR1 [pid procesu Pracownik]** do zatrzymania kolejki **oraz kill -SIGUSR2 [pid procesu Pracownik]** do wznowienia działania

## Opis Projektu

Projekt ma na celu opracowanie systemu symulującego działanie stacji narciarskiej. System obejmuje procesy obsługi klientów, symulacje narciarzy, działanie peronu stacji narciarskiej oraz Pracowników zatrzymujących stacje. Dodatkowo na koniec działania stacji generowany jest raport w plikach z informacjami o narciarzach data wejścia/wyjścia ze stacji oraz rejestry wejścia na peron.

GitHub: [https://github.com/Piotrus00/SO\\_project](https://github.com/Piotrus00/SO_project)

## Charakterystyka Projektu

- Ograniczenie ilości narciarzy na peronie
- Ograniczenie ilości narciarzy na krzesółkach
- Ograniczenie czasowe działania stacji
- Możliwość zatrzymania kolejki w razie awarii(przez sygnały)
- Synchronizacja pomiędzy Peronem, krzesółkami a narciarzem
- Zabezpieczone działania na pamięci dzielonej
- Komunikacja pomiędzy procesami
- Usuwanie/zamykanie wszystkich wykorzystywanych funkcji systemowych
- Generowanie raportu z informacjami o narciarzach

## Sposób działania

Po uruchomieniu programu **Main** tworzymy wątek odpowiedzialny za symulację czasu. Następnie przez funkcje `exe_a_proces()` uruchamiamy proces Kasjer oraz Pracownik następnie funkcja oczekuje na zakończenie tych procesów.

Utworzony **kasjer** generuje dane do karnetu(ilość dzieci narciarza, typ biletu[vip/normalny], oraz godzinę wydania karnetu) dodatkowo liczy kwotę wydanego karnetu i gotowy karnet w postaci struktury wysyła do kolejki komunikatów, gdy karnet jest gotowy uruchamiamy zostaje proces **Narciarz** który odbiera swoje dane z kolejki komunikatów i idzie na wejścia na peron dolny do 1 z 4 bramek jeżeli któraś bramka jest wolna i na obszarze peronu nie jest zbyt dużo osób żeby wejść narciarz wchodzi na teren kolejki. Teraz zależnie od jego statusu jeżeli jest vip'em to trafia do kolejki dla vip'ów priorytetowej, jeżeli jest zwykłym narciarzem trafia do kolejki normalnej.

Proces **Pracownik** odpowiadający między innymi za działanie kolejki i peronu wpuszcza na krzesółka w pierwszej kolejności vip'ów aż do zapelnienia krzesółka jeżeli nie ma vip'ów na krzesółka wpuszczamy zwykłych narciarzy gdy ustalono kto jedzie, „otwierają się bramki” do krzesółka które transportuje narciarzy na górny peron. Gdy krzesółko dojedzie wysyła wiadomość do kolejki komunikatów informując **Narciarza** że może podjąć jakąś akcje tzn. pójść na przerwę lub zjechać jedną z tras. Narciarz powtarza takie działania aż do końca swojego karnetu lub do godziny zamknięcia stacji. **Pracownik** działa do momentu zamknięcia stacji i gdy peron jest pusty **kasjer** kończy prace po opuszczeniu stoku przez wszystkich Narciarzy. Na końcu main „sprząta” po wszystkich procesach

## Mechanizmy używane przez procesy

### 1.Kasjer:

- Semaforey: gdy więcej niż 1 kasjer blokuje możliwość powielania się `id_narciarza`, przy przypisywaniu czasu do karnetu
- Kolejka komunikatów: wysyła dane Narciarza
- Pamięć dzielona : stosowane do powiększa `id_narciarza` o 1 zastosowanie przy kilku kasach, oraz dostęp do symulowanego czasu
- `Fork()` i `execl()`: do uruchamiania procesu Narciarza

## 2.Narciarz:

- Semafore: ograniczenie ilości bramek do 4, przy zapisach do plików, przy sprawdzaniu miejsca na peronie i dodawaniu osób do peronu,
- Kolejka komunikatów: jedna do odbierania danych karnetu od kasjera dwie odpowiadają za przesyłanie narciarza na peron w zależności czy jest vip'em, ostatnia odpowiada za odbieranie komunikatu gdy narciarz pojawi się na górnym peronie.
- Pamięć dzielona: stosowana do sprawdzania i dodawania osób do peronu, dostęp do symulowanego czasu.

## 3. Pracownik

- Semafore: przy odejmowaniu osób z peronu(gdy wsiadają na krzeselka), semafor do kontrolowania ilości krzesełek ustawiony na wartość 40
- Kolejka komunikatów: dwie odbierają informacje o karnecie od narciarza w zależności czy vip, jedna odpowiada za wiadomość do Narciarza, że dojechał na peron górny
- Pamięć dzielona: podobnie jak w narciarzu stosowana do sprawdzania i odejmowania osób z peronu, oraz dostęp do symulowanego czasu
- Wątki: wjazd\_krzeselek odpowiada za wjazd krzesełek

## 4.Main

Tworzy i usuwa semafore i kolejki, tworzy wątek time\_simulation\_thread odpowiedzialny za symulacje czasu

## Testy

### 1.

ustawiamy czas jazdy krzeselka na nienaturalnie długi czyli 2sekundy przy symulacji trwającej 30sekund

Pojawił się problem, main() kończył się przed dojechaniem wszystkich krzesełek i narciarze zniknęli

FIX: zmieniłem warunek pętli z

```
while (!(adres2[0] == -1 && adres[3] < 1)
```

na while(1) i dodałem fragment:

```
//przed zakonczeniem pracownik czeka az krzeselka wykonaja 1 pelny obieg zeby napewno nikt nie zostal
if (adres2[0]==-1 && adres[3]<1 && tab[0] == 0 && tab[1]==0 && tab[2]==0) {
    obieg_krzeselek++;
    if (obieg_krzeselek>=40) {
        break;
    }
}
```

Teraz wszyscy narciarze zostają przetransportowani na górę i wykonują swój zjazd a program kończy się bez błędnie.

2.

ustawiamy czas zjazdu narciarza na nienaturalnie długi czyli 2sekundy przy symulacji trwającej 30sekund

```
(kali@kali) [~/CLionProjects/untitled3]
$ ./Main
Zaczynamy!
Kajser 110662 w gotowosci
Pracownik 110663 w gotowosci!
Symulacja zakończona!
Pamiec dzielona została odłączona.
Pamiec dzielona została odłączona.
Semafor został usuniety.
Semafor został usuniety.
Semafor został usuniety.
Semafor został usuniety.
Kolejka komunikatow (ID: 32) została usunieta.
Kolejka komunikatow (ID: 33) została usunieta.
Kolejka komunikatow (ID: 34) została usunieta.
Kolejka komunikatow (ID: 35) została usunieta.
Pamiec dzielona została odłączona.
```

1	1, VIP: 0, Godzina: 08:37:45
2	2, VIP: 0, Godzina: 09:15:45
3	1, VIP: 0, Godzina: 09:18:45
4	3, VIP: 0, Godzina: 09:53:30
5	2, VIP: 0, Godzina: 09:58:00
6	1, VIP: 0, Godzina: 10:01:30
7	4, VIP: 0, Godzina: 10:30:00
8	3, VIP: 0, Godzina: 10:33:00
9	2, VIP: 0, Godzina: 10:38:45
10	1, VIP: 0, Godzina: 10:41:45
11	5, VIP: 1, Godzina: 11:07:15
12	4, VIP: 0, Godzina: 11:10:45
13	3, VIP: 0, Godzina: 11:13:45
14	2, VIP: 0, Godzina: 11:20:30
15	1, VIP: 0, Godzina: 11:22:45
16	6, VIP: 0, Godzina: 11:26:30
17	7, VIP: 0, Godzina: 11:44:30
18	5, VIP: 1, Godzina: 11:47:45
19	4, VIP: 0, Godzina: 11:51:30
20	3, VIP: 0, Godzina: 11:54:15
21	2, VIP: 0, Godzina: 12:00:45
22	1, VIP: 0, Godzina: 12:03:15
23	6, VIP: 0, Godzina: 12:06:45
24	8, VIP: 0, Godzina: 12:23:30
25	7, VIP: 0, Godzina: 12:27:00
26	5, VIP: 1, Godzina: 12:30:30
27	4, VIP: 0, Godzina: 12:34:00
28	3, VIP: 0, Godzina: 12:36:45
29	2, VIP: 0, Godzina: 12:43:45
30	9, VIP: 0, Godzina: 12:43:45
31	1, VIP: 0, Godzina: 12:45:45
32	6, VIP: 0, Godzina: 12:48:45
33	8, VIP: 0, Godzina: 13:04:45
34	7, VIP: 0, Godzina: 13:08:30
35	5, VIP: 1, Godzina: 13:11:45
36	4, VIP: 0, Godzina: 13:15:00
37	3, VIP: 0, Godzina: 13:18:00
38	10, VIP: 0, Godzina: 13:21:30
39	9, VIP: 0, Godzina: 13:25:00

wygląda na to że wszystko jest w porządku narciarze zjeżdżają a program kończy się normalnie z logów karnetu widzimy że zjazd odbywał się średnio raz na 40minut czyli prawidłowo

3.

ustawiamy limit osób na peronie na 3 (mniejszy nie ma sensu bo było by to sprzeczne z treścią i narciarz nie mógłby jechać w jednym krzeselku z dziećmi a taki jest wymóg.)

Wszystko wygląda w porządku program kończy się poprawnie logi również wyglądają sensownie.

```
└─$ ./Main
Zaczynamy!
Kajser 139180 w gotowosci
Pracownik 139181 w gotowosci!
Symulacja zakończona!
Pamiec dzielona została odłączona.
Pamiec dzielona została odłączona.
Semafor został usuniety.
Semafor został usuniety.
Semafor został usuniety.
Semafor został usuniety.
Semafor został usuniety.
Kolejka komunikatow (ID: 40) została usunieta.
Kolejka komunikatow (ID: 41) została usunieta.
Kolejka komunikatow (ID: 42) została usunieta.
Kolejka komunikatow (ID: 43) została usunieta.
Pamiec dzielona została odłączona.
```

274	2, VIP: 0, Godzina: 12:43:45
275	6, VIP: 1, Godzina: 12:45:15
276	3, VIP: 0, Godzina: 12:45:15
277	8, VIP: 0, Godzina: 12:46:15
278	1, VIP: 0, Godzina: 12:46:45
279	5, VIP: 1, Godzina: 12:47:00
280	9, VIP: 0, Godzina: 12:47:15
281	4, VIP: 0, Godzina: 12:47:45
282	2, VIP: 0, Godzina: 12:48:30
283	7, VIP: 0, Godzina: 12:48:45
284	6, VIP: 1, Godzina: 12:49:45
285	3, VIP: 0, Godzina: 12:50:15
286	8, VIP: 0, Godzina: 12:51:30
287	1, VIP: 0, Godzina: 12:52:00
288	4, VIP: 0, Godzina: 12:52:15
289	5, VIP: 1, Godzina: 12:53:15
290	9, VIP: 0, Godzina: 12:53:30
291	7, VIP: 0, Godzina: 12:54:00
292	2, VIP: 0, Godzina: 12:54:00
293	6, VIP: 1, Godzina: 12:54:00
294	3, VIP: 0, Godzina: 12:55:00
295	8, VIP: 0, Godzina: 12:56:15
296	4, VIP: 0, Godzina: 12:56:15
297	1, VIP: 0, Godzina: 12:57:30
298	5, VIP: 1, Godzina: 12:58:00
299	9, VIP: 0, Godzina: 12:58:15
300	3, VIP: 0, Godzina: 12:58:30
301	6, VIP: 1, Godzina: 12:59:45
302	10, VIP: 0, Godzina: 12:59:45
303	7, VIP: 0, Godzina: 12:59:45
304	4, VIP: 0, Godzina: 13:00:45
305	8, VIP: 0, Godzina: 13:01:30
306	1, VIP: 0, Godzina: 13:02:45
307	9, VIP: 0, Godzina: 13:03:45
308	5, VIP: 1, Godzina: 13:04:00
309	3, VIP: 0, Godzina: 13:04:45
310	7, VIP: 0, Godzina: 13:04:45
311	10, VIP: 0, Godzina: 13:04:45
312	6, VIP: 1, Godzina: 13:05:45

4.

Próba przy usunięciu wszystkich sleep'ów z programu (nie licząc czasu)

Wszystko zadziałało program wykonał się poprawnie procesy zakończyły a logi wyglądają prawidłowo

```
└─$ ./Main
```

```
Zaczynamy!
```

```
Pracownik 187070 w gotowosci!
```

```
Kajser 187069 w gotowosci
```

```
Symulacja zakończona!
```

```
Pamiec dzielona została odłączona.
```

```
Pamiec dzielona została odłączona.
```

```
Semafor został usuniety.
```

```
Semafor został usuniety.
```

```
Semafor został usuniety.
```

```
Semafor został usuniety.
```

```
Semafor został usuniety.
```

```
Kolejka komunikatow (ID: 52) została usunieta.
```

```
Kolejka komunikatow (ID: 53) została usunieta.
```

```
Kolejka komunikatow (ID: 54) została usunieta.
```

```
Kolejka komunikatow (ID: 55) została usunieta.
```

```
Pamiec dzielona została odłączona.
```



1	ID: 1, VIP: 0, Godzina: 12:00:15
2	ID: 2, VIP: 0, Godzina: 12:00:30
3	ID: 3, VIP: 0, Godzina: 16:00:45
4	ID: 4, VIP: 0, Godzina: 999:01:00
5	ID: 5, VIP: 0, Godzina: 999:01:15
6	ID: 6, VIP: 0, Godzina: 999:01:45
7	ID: 7, VIP: 1, Godzina: 12:02:15
8	ID: 8, VIP: 0, Godzina: 14:02:15
9	ID: 9, VIP: 0, Godzina: 16:02:30
10	ID: 10, VIP: 0, Godzina: 14:02:45
11	ID: 11, VIP: 0, Godzina: 16:03:15
12	ID: 12, VIP: 0, Godzina: 999:03:30
13	ID: 13, VIP: 0, Godzina: 999:03:45
14	ID: 14, VIP: 0, Godzina: 999:04:15
15	ID: 15, VIP: 0, Godzina: 12:04:30
16	ID: 16, VIP: 0, Godzina: 999:04:30
17	ID: 17, VIP: 1, Godzina: 16:05:00
18	ID: 18, VIP: 1, Godzina: 12:05:15
19	ID: 19, VIP: 0, Godzina: 12:05:30
20	ID: 20, VIP: 0, Godzina: 16:06:00
21	ID: 21, VIP: 1, Godzina: 14:06:15
22	ID: 22, VIP: 1, Godzina: 12:06:30
23	ID: 23, VIP: 0, Godzina: 12:07:00
24	ID: 24, VIP: 0, Godzina: 12:07:15
25	ID: 25, VIP: 0, Godzina: 16:07:45
26	ID: 26, VIP: 0, Godzina: 999:07:45
27	ID: 27, VIP: 0, Godzina: 16:08:00
28	ID: 28, VIP: 0, Godzina: 12:08:15
29	ID: 29, VIP: 0, Godzina: 12:08:45
30	ID: 30, VIP: 0, Godzina: 16:09:00
31	ID: 31, VIP: 0, Godzina: 14:09:15
32	ID: 32, VIP: 0, Godzina: 999:09:15
33	ID: 33, VIP: 0, Godzina: 12:09:45
34	ID: 34, VIP: 0, Godzina: 12:10:00
35	ID: 35, VIP: 0, Godzina: 999:10:15
36	ID: 36, VIP: 0, Godzina: 999:10:30
37	ID: 37, VIP: 0, Godzina: 12:11:00
38	ID: 38, VIP: 0, Godzina: 16:11:15
39	ID: 39, VIP: 0, Godzina: 12:11:30
40	ID: 40, VIP: 0, Godzina: 12:11:45

178	ID: 498, VIP: 0, <u>Godzina Karnet:</u> 14:27:00 <u>Godzina Wyjscia:</u> 14:46:15
179	ID: 416, VIP: 0, <u>Godzina Karnet:</u> 14:02:45 <u>Godzina Wyjscia:</u> 14:46:45
180	ID: 577, VIP: 0, <u>Godzina Karnet:</u> 14:48:30 <u>Godzina Wyjscia:</u> 14:48:45
181	ID: 366, VIP: 0, <u>Godzina Karnet:</u> 13:48:15 <u>Godzina Wyjscia:</u> 14:48:45
182	ID: 573, VIP: 0, <u>Godzina Karnet:</u> 14:47:15 <u>Godzina Wyjscia:</u> 14:49:00
183	ID: 501, VIP: 1, <u>Godzina Karnet:</u> 14:27:45 <u>Godzina Wyjscia:</u> 14:49:30
184	ID: 460, VIP: 0, <u>Godzina Karnet:</u> 14:16:00 <u>Godzina Wyjscia:</u> 14:51:30
185	ID: 29, VIP: 0, <u>Godzina Karnet:</u> 12:08:45 <u>Godzina Wyjscia:</u> 14:51:45
186	ID: 161, VIP: 0, <u>Godzina Karnet:</u> 12:47:00 <u>Godzina Wyjscia:</u> 14:52:00
187	ID: 526, VIP: 0, <u>Godzina Karnet:</u> 14:34:45 <u>Godzina Wyjscia:</u> 14:52:00
188	ID: 561, VIP: 0, <u>Godzina Karnet:</u> 14:44:15 <u>Godzina Wyjscia:</u> 14:52:15
189	ID: 173, VIP: 0, <u>Godzina Karnet:</u> 12:50:30 <u>Godzina Wyjscia:</u> 14:52:15
190	ID: 156, VIP: 1, <u>Godzina Karnet:</u> 12:45:45 <u>Godzina Wyjscia:</u> 14:53:30
191	ID: 541, VIP: 0, <u>Godzina Karnet:</u> 14:39:00 <u>Godzina Wyjscia:</u> 14:54:00
192	ID: 546, VIP: 0, <u>Godzina Karnet:</u> 14:40:15 <u>Godzina Wyjscia:</u> 14:54:30
193	ID: 553, VIP: 0, <u>Godzina Karnet:</u> 14:42:15 <u>Godzina Wyjscia:</u> 14:55:00
194	ID: 374, VIP: 0, <u>Godzina Karnet:</u> 13:50:30 <u>Godzina Wyjscia:</u> 14:56:45
195	ID: 545, VIP: 0, <u>Godzina Karnet:</u> 14:40:15 <u>Godzina Wyjscia:</u> 14:57:00
196	ID: 342, VIP: 0, <u>Godzina Karnet:</u> 13:41:15 <u>Godzina Wyjscia:</u> 14:57:45
197	ID: 557, VIP: 0, <u>Godzina Karnet:</u> 14:43:15 <u>Godzina Wyjscia:</u> 14:58:30
198	ID: 595, VIP: 0, <u>Godzina Karnet:</u> 14:53:45 <u>Godzina Wyjscia:</u> 14:58:45
199	ID: 493, VIP: 0, <u>Godzina Karnet:</u> 14:25:45 <u>Godzina Wyjscia:</u> 14:58:45
200	ID: 496, VIP: 0, <u>Godzina Karnet:</u> 14:26:30 <u>Godzina Wyjscia:</u> 14:59:15
201	ID: 511, VIP: 0, <u>Godzina Karnet:</u> 14:31:00 <u>Godzina Wyjscia:</u> 14:59:15
202	ID: 616, VIP: 0, <u>Godzina Karnet:</u> 15:00:00 <u>Godzina Wyjscia:</u> 15:00:00
203	ID: 598, VIP: 0, <u>Godzina Karnet:</u> 14:54:45 <u>Godzina Wyjscia:</u> 15:00:30
204	ID: 466, VIP: 0, <u>Godzina Karnet:</u> 14:18:00 <u>Godzina Wyjscia:</u> 15:00:45
205	ID: 188, VIP: 0, <u>Godzina Karnet:</u> 12:55:15 <u>Godzina Wyjscia:</u> 15:01:00
206	ID: 425, VIP: 0, <u>Godzina Karnet:</u> 14:05:15 <u>Godzina Wyjscia:</u> 15:01:00
207	ID: 586, VIP: 0, <u>Godzina Karnet:</u> 14:51:00 <u>Godzina Wyjscia:</u> 15:01:15
208	ID: 192, VIP: 1, <u>Godzina Karnet:</u> 14:56:30 <u>Godzina Wyjscia:</u> 15:02:30
209	ID: 115, VIP: 0, <u>Godzina Karnet:</u> 14:33:30 <u>Godzina Wyjscia:</u> 15:02:30
210	ID: 157, VIP: 0, <u>Godzina Karnet:</u> 12:46:00 <u>Godzina Wyjscia:</u> 15:02:45
211	ID: 489, VIP: 0, <u>Godzina Karnet:</u> 14:24:30 <u>Godzina Wyjscia:</u> 15:03:15
212	ID: 191, VIP: 0, <u>Godzina Karnet:</u> 14:56:00 <u>Godzina Wyjscia:</u> 15:04:30
213	ID: 413, VIP: 0, <u>Godzina Karnet:</u> 14:02:00 <u>Godzina Wyjscia:</u> 15:04:45
214	ID: 182, VIP: 0, <u>Godzina Karnet:</u> 14:53:30 <u>Godzina Wyjscia:</u> 15:05:30
215	ID: 560, VIP: 0, <u>Godzina Karnet:</u> 14:44:00 <u>Godzina Wyjscia:</u> 15:06:45
216	ID: 467, VIP: 0, <u>Godzina Karnet:</u> 14:18:15 <u>Godzina Wyjscia:</u> 15:07:45
217	ID: 600, VIP: 0, <u>Godzina Karnet:</u> 14:55:30 <u>Godzina Wyjscia:</u> 15:08:30
218	ID: 528, VIP: 0, <u>Godzina Karnet:</u> 14:35:15 <u>Godzina Wyjscia:</u> 15:08:30

5.

sprawdzamy zachowanie w sytuacji „dużej awarii” wysyłamy sygnał do zatrzymania kolejki czekamy na zakończenie symulacji czasu i wznawiamy kolejkę żeby zobaczyć co się wydarzy.

```
└─$ ./Main
Zaczynamy!
Kajser 11608 w gotowosci
Pracownik 11609 w gotowosci!
[prac1] Otrzymano SIGUSR1 -> Zatrzymuję kolejkę.
Symulacja zakończona!
[prac2] Otrzymano SIGUSR2 -> Wznawiam kolejkę.
Pamiec dzielona została odłączona.
Pamiec dzielona została odłączona.
Semafor został usuniety.
Semafor został usuniety.
Semafor został usuniety.
Semafor został usuniety.
Kolejka komunikatow (ID: 8) została usunieta.
Kolejka komunikatow (ID: 9) została usunieta.
Kolejka komunikatow (ID: 10) została usunieta.
Kolejka komunikatow (ID: 11) została usunieta.
Pamiec dzielona została odłączona.

(kali@kali) - [~/CLionProjects/untitled3]
└─$
```




Wszystko zadziało poprawnie narciarze dojechali na górny peron i zjechali po czym opuścili stacje

6.

sytuacja gdy na peronie mamy jednego normalnego narciarza i samych vip'ow

pojawił się problem, za zaczęto udaje mu się dostać do krzeselka pomimo dużej ilości vip'ow

FIX: krzeselka nie miały opóźnień i jechały zbyt krótko w stosunku do zjazdu przez co nie tworzyły się kolejki zmieniłem długość jazdy krzeselka teraz wszystko jest bardziej naturalne czyli normalnemu przez cały dzień udało się zjechać tylko kilka razy (dlatego że pojawił się jako 1-wszy i nie było kolejek, w godzinach szczytu nie zjechał ani razu) C1=1111 to nasz podstawiony narciarz 16 to przykładowy narciarz z karnetem na 24 godziny

	C1 	C2 	C3 
4	16	VIP: 1	Godzina: 09:29:45
5	1111	VIP: 0	Godzina: 09:30:30
6	16	VIP: 1	Godzina: 10:12:15
7	1111	VIP: 0	Godzina: 10:14:00
8	16	VIP: 1	Godzina: 10:36:00
9	1111	VIP: 0	Godzina: 10:56:30
10	16	VIP: 1	Godzina: 11:17:45
11	1111	VIP: 0	Godzina: 11:20:30
12	16	VIP: 1	Godzina: 11:41:45
13	1111	VIP: 0	Godzina: 12:04:30
14	16	VIP: 1	Godzina: 12:22:45
15	16	VIP: 1	Godzina: 12:47:00
16	1111	VIP: 0	Godzina: 12:47:00
17	16	VIP: 1	Godzina: 13:28:30
18	1111	VIP: 0	Godzina: 13:28:30
19	16	VIP: 1	Godzina: 13:50:45
20	16	VIP: 1	Godzina: 14:14:45
21	16	VIP: 1	Godzina: 14:55:45
22	16	VIP: 1	Godzina: 15:36:00
23	16	VIP: 1	Godzina: 15:59:15
24	16	VIP: 1	Godzina: 16:35:30
25	16	VIP: 1	Godzina: 16:57:45
26	16	VIP: 1	Godzina: 17:38:30
27	16	VIP: 1	Godzina: 18:36:30
28	16	VIP: 1	Godzina: 20:10:30
29	16	VIP: 1	Godzina: 20:48:00
30	1111	VIP: 0	Godzina: -1:-1:-1

### Przypadek gdy nie przyjdzie żaden Narciarz

Wszystko działa poprawnie krzesółka jeżdżą do momentu zakończenia się czasu działania stacji.

## Trudności napotkane podczas pracy nad projektem

- Największy problem sprawiało odczytywanie wysłanych karnetów przez narciarzy początkowo problem ten próbowałem rozwiązać semaforami i pamięcią dzieloną jednak nie udało mi się tego osiągnąć bardzo często występowały deadlock'i lub adres nie został odczytany przez narciarza i proces nie mógł się skończyć ostatecznie rozwiązałem ten problem korzystając z kolejki komunikatów w której każdy narciarz ma swój własny mtype którym przychodzi komunikat, że jest na górze stacji i może wykonywać dalej swoje akcje.
- Innym problemem, na który poświęciłem dużo czasu było umieszczanie vip'ów na krzeselkach poza kolejką tutaj głównym problemem był pomysł jak to wykonać oraz problem związany z tym, że vip'y działały w pętli co w pewnym momencie pracy zaburzyło całe działanie mojego programu
- Trudnością bardziej ogólną stanowiło dla mnie ilość uwagi i skupienia które trzeba było włożyć podczas pisania projektu, w momencie popełnienia jakiegoś małego błędu typu zły klucz w semaforze potrafiłem stracić nawet kilka godzin, żeby dopiero zauważyć błąd
- Innym problemem związanym z projektem było znajdowanie swoich błędów do tego wykorzystywałem debugger który czasami albo nie pomagał albo nawet utrudniał znalezienie błędu.

## Linki do wybranych elementów kodu

### 1. Main

Tworzenie kolejek komunikatów:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/main.c#L21](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/main.c#L21)

tworzenie semaforów:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/main.c#L51-L55](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/main.c#L51-L55)

Inicjalizacja pamięci dzielonej:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/main.c#L111-L112](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/main.c#L111-L112)

tworzenie wątku:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/main.c#L58C1-L62C6](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/main.c#L58C1-L62C6)

przykładowe użycie semaforów i pamięci dzielonej:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/main.c#L122-L126](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/main.c#L122-L126)

usuwanie semaforów:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/main.c#L81-L85](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/main.c#L81-L85)

usuwanie kolejek komunikatów:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/main.c#L87-L90](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/main.c#L87-L90)

odłączanie pamięci:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/main.c#L91](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/main.c#L91)

fork() i execl():

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/main.c#L97C1-L101C6](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/main.c#L97C1-L101C6)

## 2.Kasjer

Inicjalizacja pamięci dzielonej:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Kasjer.c#L52C1-L55C12](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Kasjer.c#L52C1-L55C12)

Dodawanie semaforów:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Kasjer.c#L58C1-L59C51](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Kasjer.c#L58C1-L59C51)

Przyłączanie kolejki komunikatów:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Kasjer.c#L61C1-L65C6](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Kasjer.c#L61C1-L65C6)

Przykładowe zastosowanie semaforów i pamięci dzielonej:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Kasjer.c#L95C1-L104C31](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Kasjer.c#L95C1-L104C31)

Wysyłanie komunikatu:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Kasjer.c#L134](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Kasjer.c#L134)

Obsługa sygnału:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Kasjer.c#L41](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Kasjer.c#L41)

Odłączanie pamięci dzielonej:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Kasjer.c#L148](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Kasjer.c#L148)

### 3.Pracownik

Inicjalizacja pamięci dzielonej:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L69C1-L72C12](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L69C1-L72C12)

Dodawanie semaforów:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L79C1-L83C46](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L79C1-L83C46)

Przyłączenie kolejki komunikatów:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L74C1-L76C59](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L74C1-L76C59)

Przykładowy odczyt z kolejki:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L106](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L106)

Tworzenie wątku krzesełek:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L173C1-L177C10](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L173C1-L177C10)

Mutex:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L16](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L16)

Zastosowanie mutexu:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L224C1-L241C40](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L224C1-L241C40)

Obsługa sygnałów:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L59-L60](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L59-L60)

Przykładowe zastosowanie semaforów i pamięci dzielonej:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L154C1-L156C34](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L154C1-L156C34)

Zastosowanie semaforów jako blokady ilości:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L171C1-L171C41](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L171C1-L171C41)



Zapis do pliku:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L186C1-L194C2](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Pracownik.c#L186C1-L194C2)

### 3.Narciarz

Inicjalizacja pamięci dzielonej:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Narciarz.c#L59C1-L62C12](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Narciarz.c#L59C1-L62C12)

Dodawanie semaforów:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Narciarz.c#L74C1-L75C63](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Narciarz.c#L74C1-L75C63)

Przyłączenie kolejki komunikatów:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Narciarz.c#L68C1-L71C59](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Narciarz.c#L68C1-L71C59)

Przykładowy odczyt z kolejki:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Narciarz.c#L78C1-L81C6](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Narciarz.c#L78C1-L81C6)

Przykładowe zastosowanie semaforów i pamięci dzielonej:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Narciarz.c#L120C1-L129C11](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Narciarz.c#L120C1-L129C11)

Zastosowanie semaforów jako blokady ilości:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Narciarz.c#L117](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Narciarz.c#L117)

Zapis do pliku:

[https://github.com/Piotrus00/SO\\_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Narciarz.c#L208C1-L214C18](https://github.com/Piotrus00/SO_project/blob/15854f21d8bc40b590f0c8cd236ea72aff5e9afd/Narciarz.c#L208C1-L214C18)