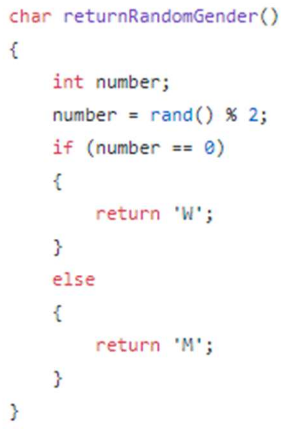
Task 5 – Compte rendu – De Araujo Antoine

Part1 – Preseantation

Instructions :
The analogy is based upon a hypothetical barber shop with many barbers serving women and man. In a barber shop there are N1 barbers serving only women, N2 barbers serving only men and N3 barbers serving both women and men. Each barber has one barber's chair in a cutting room. In a waiting room there are M chairs. When the barber finishes cutting a customer's hair, he dismisses the customer and goes to the waiting room to see if there are others waiting. If there are, he brings one of them (but only if he is able to cut hair, i.e. the barber brings a man into the cutting room only if he can cut his hair, etc.) back to the chair and cuts hair (it lasts a random time). If there are none, he returns to the chair and sleeps in it. Each customer can be either a male client or a female client. When a client arrives, he/she checks what barbers are doing. If there is any barber sleeping (who is able to serve the client), the customer wakes him up, and sits in the cutting room chair. If all barbers (able to serve the client) are cutting hair, the customer stays in the waiting room. If there is a free chair in the waiting room, the customer sits in it and waits their turn. If there is no free chair, the customer leaves. Implement the C language the sleeping barber problem.
The program must be parameterized by N1, N2, N3 - number of barbers, M - number of chairs in waiting room. Print on the standard output verbose messages from customers and the barbers. Execute the program with different N1, N2, N3 and M values.
Describe a solution for this problem in a document (uploaded to github). Verify your program with different N1, N2, N3 and M values.

Part2 – Code presentation

I – Some functions useful for my main prgram.

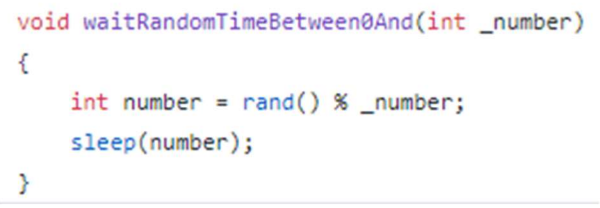


```c
char returnRandomGender()
{
    int number;
    number = rand() % 2;
    if (number == 0)
    {
        return 'W';
    }
    else
    {
        return 'M';
    }
}
```

This function called returnRandomGender() permit to define the sex of a new client (Women or Men), using the rand() function and by returning the sex of the client.

W = women

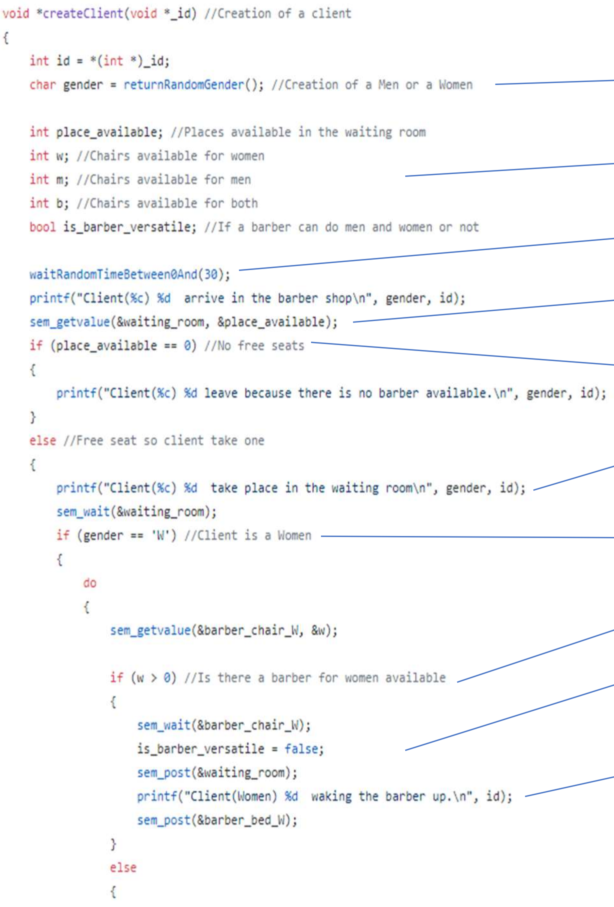
M = men



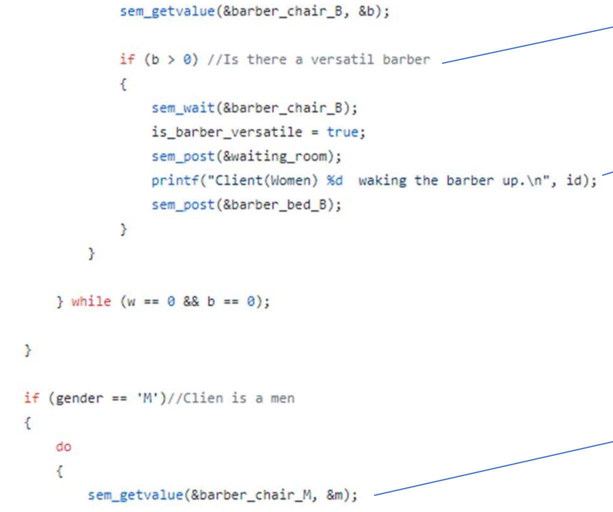
```c
void waitRandomTimeBetween0And(int _number)
{
    int number = rand() % _number;
    sleep(number);
}
```

This function called waitRandomTimeBetween0And(int) permit to generate a random time before the arrival of the first client.

II – Creation of my client



```c
void *createClient(void *_id) //Creation of a client
{
    int id = *(int *)_id;
    char gender = returnRandomGender(); //Creation of a Men or a Women

    int place_available; //Places available in the waiting room
    int w; //Chairs available for women
    int m; //Chairs available for men
    int b; //Chairs available for both
    bool is_barber_versatile; //If a barber can do men and women or not

    waitRandomTimeBetween0And(30);
    printf("Client(%c) %d  arrive in the barber shop\n", gender, id);
    sem_getvalue(&waiting_room, &place_available);
    if (place_available == 0) //No free seats
    {
        printf("Client(%c) %d leave because there is no barber available.\n", gender, id);
    }
    else //Free seat so client take one
    {
        printf("Client(%c) %d  take place in the waiting room\n", gender, id);
        sem_wait(&waiting_room);
        if (gender == 'W') //Client is a Women
        {
            do
            {
                sem_getvalue(&barber_chair_W, &w);

                if (w > 0) //Is there a barber for women available
                {
                    sem_wait(&barber_chair_W);
                    is_barber_versatile = false;
                    sem_post(&waiting_room);
                    printf("Client(Women) %d  waking the barber up.\n", id);
                    sem_post(&barber_bed_W);
                }
                else
                {
```



```c
                    sem_getvalue(&barber_chair_B, &b);

                    if (b > 0) //Is there a versatil barber
                    {
                        sem_wait(&barber_chair_B);
                        is_barber_versatile = true;
                        sem_post(&waiting_room);
                        printf("Client(Women) %d  waking the barber up.\n", id);
                        sem_post(&barber_bed_B);
                    }
                }

            } while (w == 0 && b == 0);

        }

        if (gender == 'M')//Clien is a men
        {
            do
            {
                sem_getvalue(&barber_chair_M, &m);
```

Starting by defining the gender of my client with the function.

Initialisation of my variables.

Waiting time before this client come to the barbershop using the function.

Client arriving at the barber shop and then checking if they are free seats.

If not client is leaving

If yes client took a seat then wait

If client is a Women

Looking for a barber taking care of womens

If barber for women available

Assign this barber to this client

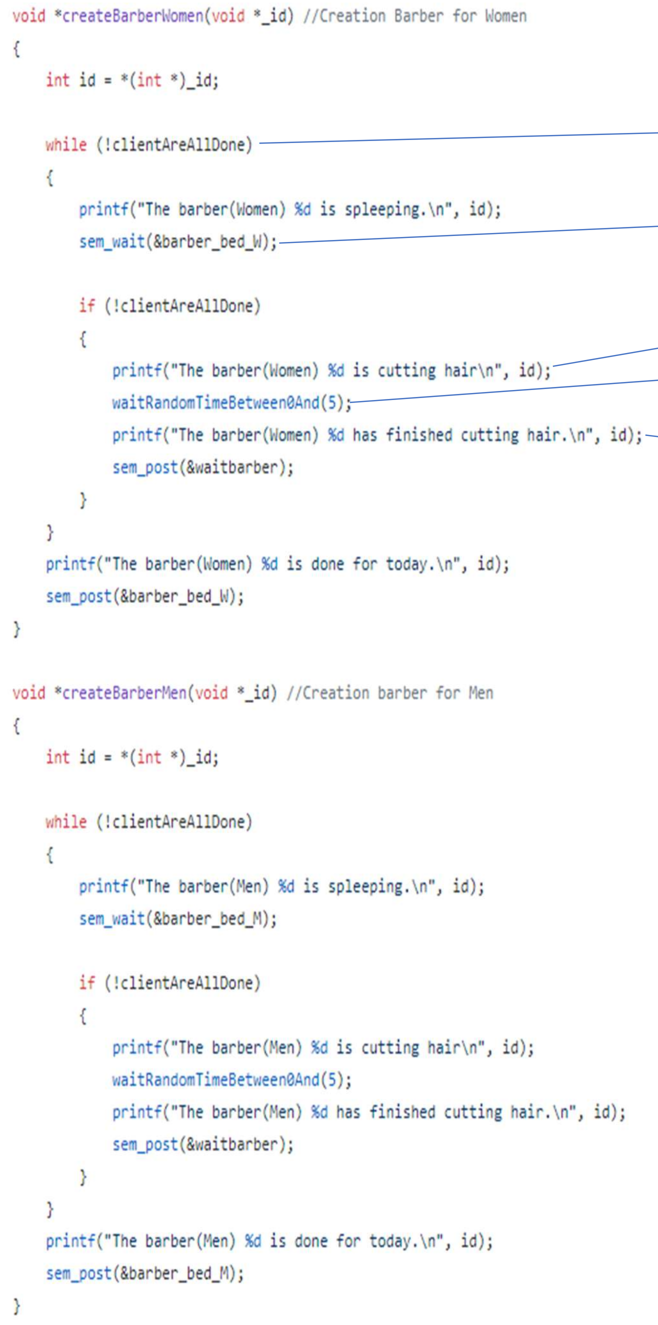
If not, looking if there is a versatil barber available

Assign this barber to this client

Then the thing happend if the client is a men.

III – Creation of a barber

So there is three type of barbers ; the one that can take care of men, the one of women and the last one can take care of both.



```
void *createBarberWomen(void *_id) //Creation Barber for Women
{
    int id = *(int *)_id;

    while (!clientAreAllDone)
    {
        printf("The barber(Women) %d is spleeping.\n", id);
        sem_wait(&barber_bed_W);

        if (!clientAreAllDone)
        {
            printf("The barber(Women) %d is cutting hair\n", id);
            waitRandomTimeBetween0And(5);
            printf("The barber(Women) %d has finished cutting hair.\n", id);
            sem_post(&waitbarber);
        }
    }
    printf("The barber(Women) %d is done for today.\n", id);
    sem_post(&barber_bed_W);
}

void *createBarberMen(void *_id) //Creation barber for Men
{
    int id = *(int *)_id;

    while (!clientAreAllDone)
    {
        printf("The barber(Men) %d is spleeping.\n", id);
        sem_wait(&barber_bed_M);

        if (!clientAreAllDone)
        {
            printf("The barber(Men) %d is cutting hair\n", id);
            waitRandomTimeBetween0And(5);
            printf("The barber(Men) %d has finished cutting hair.\n", id);
            sem_post(&waitbarber);
        }
    }
    printf("The barber(Men) %d is done for today.\n", id);
    sem_post(&barber_bed_M);
}
```

While all the clients are not done the process continue.

Waiting for a client

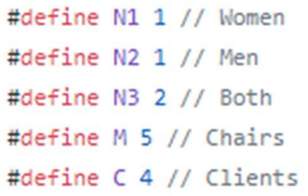
Barber cutting hair to his client

Random time corresponding at the time it take to take care of the client hairs.

Then he finish

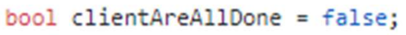
Same thing for other barbers.

IV – Main program



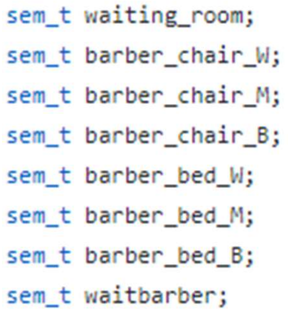
```
#define N1 1 // Women
#define N2 1 // Men
#define N3 2 // Both
#define M 5 // Chairs
#define C 4 // Clients
```

This defines are corresponding to the mains variables of my program. The number at the right of the variable name is the number of each ones I want to create. For the chairs this is 'M' and I want 5 of them.



```
bool clientAreAllDone = false;
```

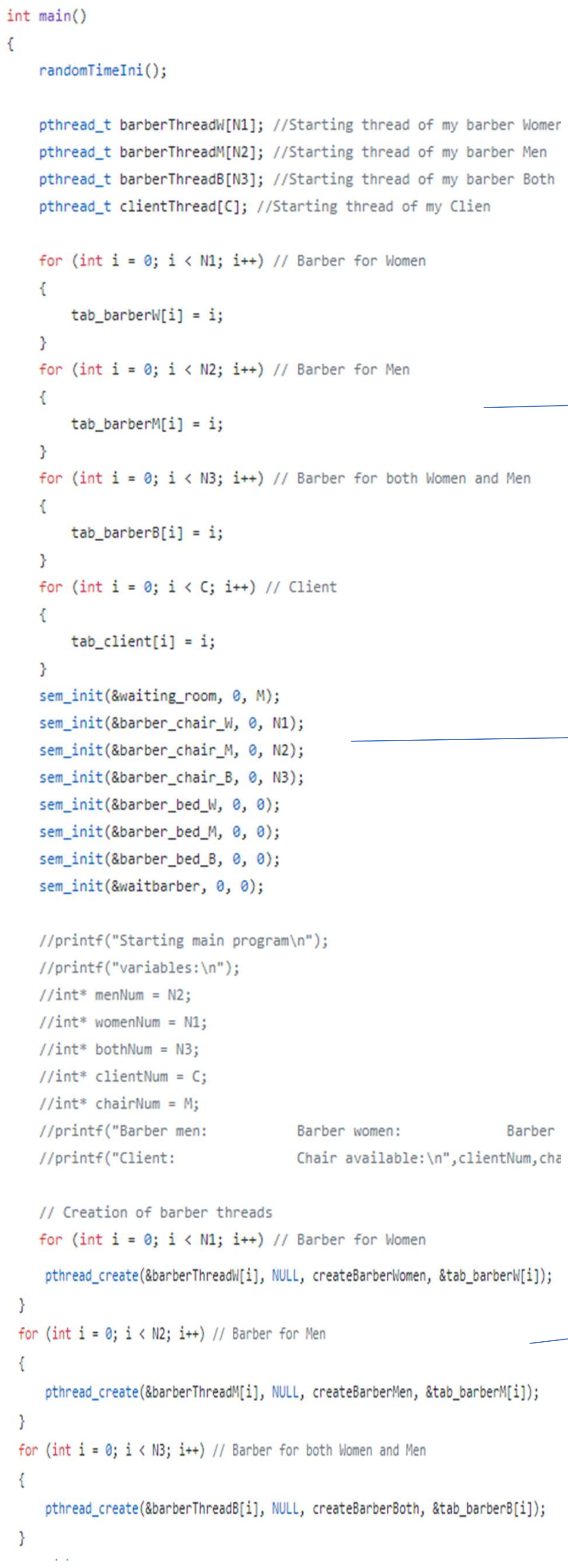
At the begining of my program my booelan clientAreAllDone need to bet set on false.



```
sem_t waiting_room;
sem_t barber_chair_W;
sem_t barber_chair_M;
sem_t barber_chair_B;
sem_t barber_bed_W;
sem_t barber_bed_M;
sem_t barber_bed_B;
sem_t waitbarber;
```

I'm creating all my semaphore necessary

Then I can start my main.



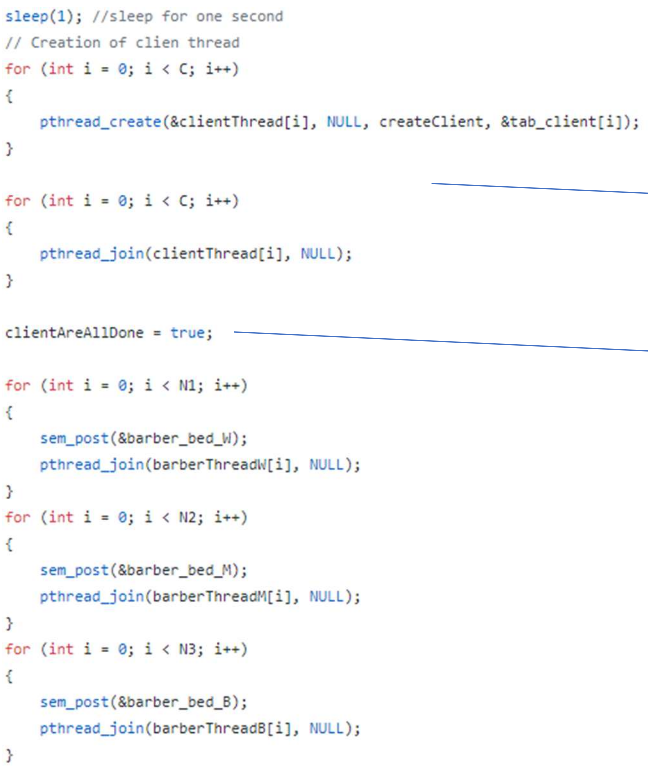
```c
int main()
{
    randomTimeIni();

    pthread_t barberThreadW[N1]; //Starting thread of my barber Womer
    pthread_t barberThreadM[N2]; //Starting thread of my barber Men
    pthread_t barberThreadB[N3]; //Starting thread of my barber Both
    pthread_t clientThread[C]; //Starting thread of my Clien

    for (int i = 0; i < N1; i++) // Barber for Women
    {
        tab_barberW[i] = i;
    }
    for (int i = 0; i < N2; i++) // Barber for Men
    {
        tab_barberM[i] = i;
    }
    for (int i = 0; i < N3; i++) // Barber for both Women and Men
    {
        tab_barberB[i] = i;
    }
    for (int i = 0; i < C; i++) // Client
    {
        tab_client[i] = i;
    }
    sem_init(&waiting_room, 0, M);
    sem_init(&barber_chair_W, 0, N1);
    sem_init(&barber_chair_M, 0, N2);
    sem_init(&barber_chair_B, 0, N3);
    sem_init(&barber_bed_W, 0, 0);
    sem_init(&barber_bed_M, 0, 0);
    sem_init(&barber_bed_B, 0, 0);
    sem_init(&waitbarber, 0, 0);

    //printf("Starting main program\n");
    //printf("variables:\n");
    //int* menNum = N2;
    //int* womenNum = N1;
    //int* bothNum = N3;
    //int* clientNum = C;
    //int* chairNum = M;
    //printf("Barber men:           Barber women:           Barber
    //printf("Client:               Chair available:\n",clientNum,cha

    // Creation of barber threads
    for (int i = 0; i < N1; i++) // Barber for Women
      pthread_create(&barberThreadW[i], NULL, createBarberWomen, &tab_barberW[i]);
  }
  for (int i = 0; i < N2; i++) // Barber for Men
  {
      pthread_create(&barberThreadM[i], NULL, createBarberMen, &tab_barberM[i]);
  }
  for (int i = 0; i < N3; i++) // Barber for both Women and Men
  {
      pthread_create(&barberThreadB[i], NULL, createBarberBoth, &tab_barberB[i]);
  }
```

Starting all my thread

Creation of the number I ask of each objects

Initialisation of my semaphores

Creation of my barbers threads



```
sleep(1); //sleep for one second
// Creation of clien thread
for (int i = 0; i < C; i++)
{
    pthread_create(&clientThread[i], NULL, createClient, &tab_client[i]);
}

for (int i = 0; i < C; i++)
{
    pthread_join(clientThread[i], NULL);
}

clientAreAllDone = true;

for (int i = 0; i < N1; i++)
{
    sem_post(&barber_bed_W);
    pthread_join(barberThreadW[i], NULL);
}
for (int i = 0; i < N2; i++)
{
    sem_post(&barber_bed_M);
    pthread_join(barberThreadM[i], NULL);
}
for (int i = 0; i < N3; i++)
{
    sem_post(&barber_bed_B);
    pthread_join(barberThreadB[i], NULL);
}
```

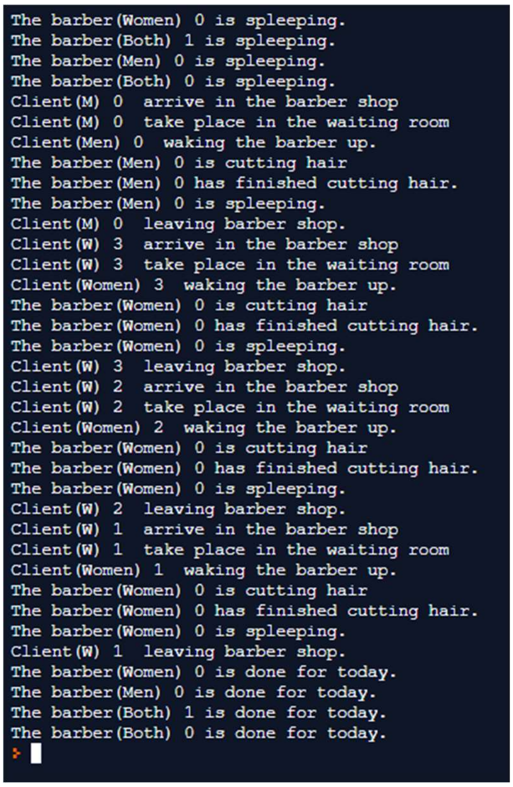
Creation of my clients threads

Switching my boolean clientAreAllDone to true

Part3 – Exemples

I - Basic Exemple

Barber : 1 for Women, for 1Men, 2 for Both / 5 chairs / 4 Clients



```
The barber(Women) 0 is spleeping.
The barber(Both) 1 is spleeping.
The barber(Men) 0 is spleeping.
The barber(Both) 0 is spleeping.
Client(M) 0  arrive in the barber shop
Client(M) 0  take place in the waiting room
Client(Men) 0  waking the barber up.
The barber(Men) 0 is cutting hair
The barber(Men) 0 has finished cutting hair.
The barber(Men) 0 is spleeping.
Client(M) 0  leaving barber shop.
Client(W) 3  arrive in the barber shop
Client(W) 3  take place in the waiting room
Client(Women) 3  waking the barber up.
The barber(Women) 0 is cutting hair
The barber(Women) 0 has finished cutting hair.
The barber(Women) 0 is spleeping.
Client(W) 3  leaving barber shop.
Client(W) 2  arrive in the barber shop
Client(W) 2  take place in the waiting room
Client(Women) 2  waking the barber up.
The barber(Women) 0 is cutting hair
The barber(Women) 0 has finished cutting hair.
The barber(Women) 0 is spleeping.
Client(W) 2  leaving barber shop.
Client(W) 1  arrive in the barber shop
Client(W) 1  take place in the waiting room
Client(Women) 1  waking the barber up.
The barber(Women) 0 is cutting hair
The barber(Women) 0 has finished cutting hair.
The barber(Women) 0 is spleeping.
Client(W) 1  leaving barber shop.
The barber(Women) 0 is done for today.
The barber(Men) 0 is done for today.
The barber(Both) 1 is done for today.
The barber(Both) 0 is done for today.
> 
```

At the begining all the barbers are sleeping waiting for the first client

The first client ; a men arrive and he take place in the waiting room (4 seats available now). The barber for men is taken in priority and starting cuting the men client’s hair. He finish it and go back waiting another client. The first client leaving.

Same thing happend with a women client and the barber taking care of women.

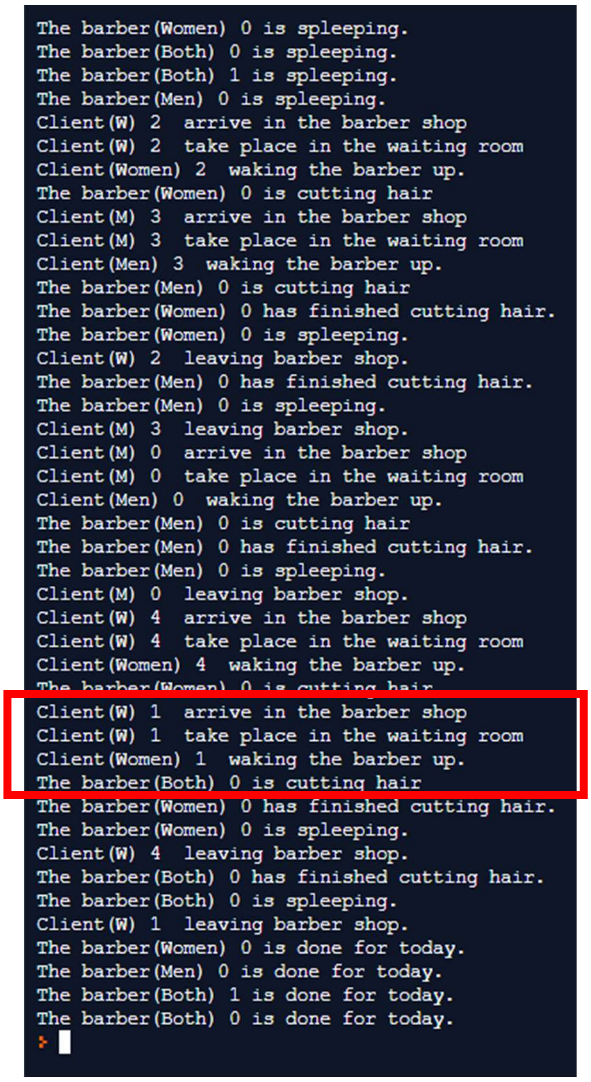
Same thing happend with a women client and the barber taking care of women.

Same thing happend with a women client and the barber taking care of women.

There 4 clients have been made so the day is over and all the barbers are done

II – Exemple with BarberBoth needed

Barber : 1 for Women, for 1Men, 2 for Both / 3 chairs / 4 Clients

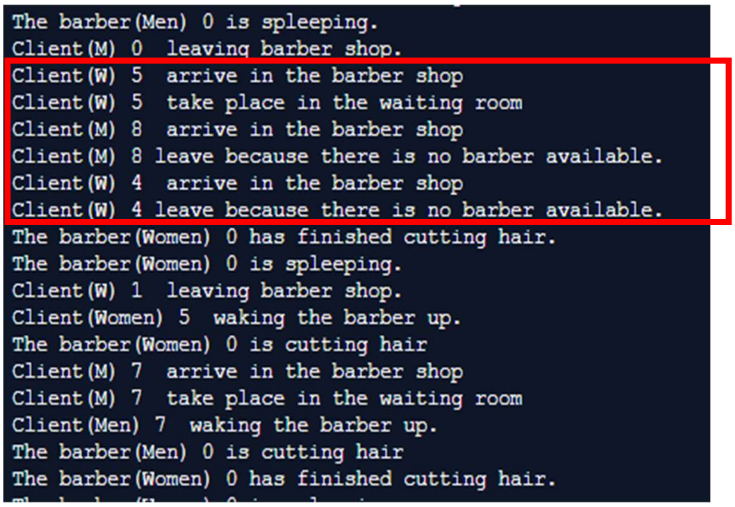


```
The barber(Women) 0 is spleeping.
The barber(Both) 0 is spleeping.
The barber(Both) 1 is spleeping.
The barber(Men) 0 is spleeping.
Client(W) 2  arrive in the barber shop
Client(W) 2  take place in the waiting room
Client(Women) 2  waking the barber up.
The barber(Women) 0 is cutting hair
Client(M) 3  arrive in the barber shop
Client(M) 3  take place in the waiting room
Client(Men) 3  waking the barber up.
The barber(Men) 0 is cutting hair
The barber(Women) 0 has finished cutting hair.
The barber(Women) 0 is spleeping.
Client(W) 2  leaving barber shop.
The barber(Men) 0 has finished cutting hair.
The barber(Men) 0 is spleeping.
Client(M) 3  leaving barber shop.
Client(M) 0  arrive in the barber shop
Client(M) 0  take place in the waiting room
Client(Men) 0  waking the barber up.
The barber(Men) 0 is cutting hair
The barber(Men) 0 has finished cutting hair.
The barber(Men) 0 is spleeping.
Client(M) 0  leaving barber shop.
Client(W) 4  arrive in the barber shop
Client(W) 4  take place in the waiting room
Client(Women) 4  waking the barber up.
The barber(Women) 0 is cutting hair
Client(W) 1  arrive in the barber shop
Client(W) 1  take place in the waiting room
Client(Women) 1  waking the barber up.
The barber(Both) 0 is cutting hair
The barber(Women) 0 has finished cutting hair.
The barber(Women) 0 is spleeping.
Client(W) 4  leaving barber shop.
The barber(Both) 0 has finished cutting hair.
The barber(Both) 0 is spleeping.
Client(W) 1  leaving barber shop.
The barber(Women) 0 is done for today.
The barber(Men) 0 is done for today.
The barber(Both) 1 is done for today.
The barber(Both) 0 is done for today.
> 
```

In this case we have our two barber who can take care of only women and only men that are already occuped with clients. So when the Women client (1) enter in the barber shop this is the barber that can take care of women and men who is going to be requested.

III – Exemple with no barber available

Barber : 1 for Women, for 1Men, 0 for Both / 1 chairs / 10 Clients



```
The barber(Men) 0 is spleeping.
Client(M) 0  leaving barber shop.
Client(W) 5  arrive in the barber shop
Client(W) 5  take place in the waiting room
Client(M) 8  arrive in the barber shop
Client(M) 8 leave because there is no barber available.
Client(W) 4  arrive in the barber shop
Client(W) 4 leave because there is no barber available.
The barber(Women) 0 has finished cutting hair.
The barber(Women) 0 is spleeping.
Client(W) 1  leaving barber shop.
Client(Women) 5  waking the barber up.
The barber(Women) 0 is cutting hair
Client(M) 7  arrive in the barber shop
Client(M) 7  take place in the waiting room
Client(Men) 7  waking the barber up.
The barber(Men) 0 is cutting hair
The barber(Women) 0 has finished cutting hair.
```

In this exemple both barbers are already occuped with client, and the place in the waiting room is taken by the client 5. So when the client 8 and 4 came at the barber shop no free seats where available and they had to leave.