

Aluno: Victor Pessoa Oliveira Ortins

Matrícula: 20210024667

Github gist (Código mais legível):

<https://gist.github.com/VictorOrtins/fcdc933fb2aef6919c7c00e01be00aeb>

Prints do código:

```
 1  main.cpp > [o] frases
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12 const int QTD_THREADS_CONSUMIDORAS = 4; //Quantidade de threads consumidoras
13 const int QTD_THREADS_PRODUTORAS = 4; //Quantidade de threads produtoras
14
15 const vector<string> frases{ "Arsenal Liverpool Manchester City", "Aston Villa Tottenham Manchester United",
16                                "West Ham Brighton Wolves", "Newcastle Chelsea Fulham Bournemouth",
17                                "Crystal Palace Brentford Everton",
18                                "Nottingham Forest Luton Town Burnley Sheffield United"};
19                                //Frases a serem enviadas (tabela da Premier League quando enviei a primeira atv)
20
21 vector<string> buffer; //Vector que serve como um buffer
22 bool consumidor_leu[QTD_THREADS_CONSUMIDORAS]; //Vetor para indicar qual consumidor leu a mensagem atual (a primeira)
23
24 sem_t mutex_buffer; //Mutex para o acesso ao buffer
25 sem_t mutex_leu; //Mutex para o acesso a variável consumidor_leu
```

```
27 void producer(int i){
28     int index_frase = i % frases.size(); //Pega a frase que será mandada pelo produtor
29     //Como a quantidade de threads produtoras pode ser maior que a quantidade de frases possíveis, utilizo
30     //o mod para garantir que sempre será pego alguma frase válida no vector
31
32     string frase = frases[index_frase]; //Pegar a frase
33
34     stringstream ss(frase); //Inicializando variável que fará basicamente o split da mensagem em palavras
35
36     string token; //Token que armazenará cada palavra da mensagem
37
38     int contador = 0; //Inicialização do contador para saber se será feito um push back no buffer
39     //Ou se será apenas alterado a string no buffer
40
41
42     sem_wait(&mutex_buffer); //Mutex para garantir que o buffer está apenas sendo acessado por uma thread
43     //Ele é colocado mesmo antes do while para impedir que as mensagens das threads fiquem embaralhadas
44     cout << "Thread produtora de id " << i << " enviará mensagem: " << frase << "\n";
45
46     while(ss >> token){ //Enquanto ainda houver palavras na stringstream
47         if (contador == 0){ //Se for a primeira palavra
48             cout << "Thread produtora de id " << i << " colocou no buffer: " << token << "\n";
49             buffer.push_back(token); //Push back no buffer (nova mensagem está sendo inserida)
50             contador++; //Indicando que a primeira palavra já foi
51             continue;
52         }
53         cout << "Thread produtora de id " << i << " colocou no buffer: " << token << "\n";
54
55         buffer.at(buffer.size() - 1) = buffer.at(buffer.size() - 1) + ' ' + token; //Pegando a última palavra
56         //adicionada (que foi adicionada por essa thread) e adicionada a próxima palavra
57         sleep(1);
58     }
59     sem_post(&mutex_buffer); //Liberando a seção crítica
```

```

1 void consumer(int i){
2     int id = gettid(); //Pegando o thread id
3
4     while(!buffer.empty()){ //Enquanto ainda existirem palavras a serem lidas
5         sem_wait(&mutex_leu); //Entrando na seção crítica da variável consumidor_leu
6
7         bool terminou = true; //Se todos os consumidores leram, terminou
8         for(int i = 0; i < QTD_THREADS_CONSUMIDORAS; i++){
9             if(consumidor_leu[i] == false){ //Se algum consumidor não leu
10                 terminou = false; //Não terminou
11                 break;
12             }
13         }
14
15         if(terminou){ //Se terminou
16             sem_wait(&mutex_buffer); //Entrando na seção crítica do buffer
17             //Ou seja, não dá pra dar esse erase se algum produtor estiver produzindo
18             //Ou se algum consumidor estiver consumindo
19             buffer.erase(buffer.begin()); //Apaga a primeira mensagem do buffer
20             sem_post(&mutex_buffer); //Saindo da seção crítica do buffer
21
22             //Além disso, setando todos os consumidor_leu para falso a fim de liberar os consumidores
23             for(int i = 0; i < QTD_THREADS_CONSUMIDORAS; i++){
24                 consumidor_leu[i] = false;
25             }
26         }
27     else{
28         if(consumidor_leu[i] == true){ //Se o consumidor atual já leu
29             sem_post(&mutex_leu); //Libera o mutex da variável consumidor_leu
30             continue; //Não vai conseguir ler a mensagem no buffer
31         }
32     }
33     sem_post(&mutex_leu); //Liberando o mutex de consumidor_leu

```

```

sem_post(&mutex_leu); //Liberando o mutex de consumidor_leu

sem_wait(&mutex_buffer); //Entrando na seção crítica do buffer
if(!buffer.empty()){ //Se o estiver vazio
    string mensagem = buffer.front(); //Pegando a primeira mensagem do buffer
    cout << "Thread de tid " << id << " leu mensagem: " << mensagem << "\n";

    sem_wait(&mutex_leu); //Entrando na seção crítica de consumidor_leu
    //Isso garante que não vai haver nenhuma thread que está consumindo do buffer alterando
    //a variável consumidor_leu enquanto algum consumidor está checando se todas as threads já leram do
    //buffer
    consumidor_leu[i] = true;
    sem_post(&mutex_leu); //Saindo da seção crítica de consumidor_leu

    sleep(1);
}

sem_post(&mutex_buffer); //Saindo da seção crítica do buffer
}
}

```

```

int main(void){
    vector<thread> threads_produtoras; //Criação do vector de threads produtoras
    vector<thread> threads_consumidoras; //Criação do vector de threads consumidoras

    //Como os 2 são mutexes, preciso inicializá-los com 1, para que o acesso seja mutualmente exclusivo
    sem_init(&mutex_buffer, 0, 1); //Inicialização do mutex do buffer
    sem_init(&mutex_leu, 0, 1); //Inicialização do mutex do consumidor_leu

    for(int i = 0; i < QTD_THREADS_PRODUTORAS; i++){
        std::cout << "Thread produtora de id " << i << " iniciando\n";
        thread produtora(producer, i); //Início da thread produtora
        threads_produtoras.emplace_back(std::move(produtora)); //Colocando no vector de threads produtoras
    }

    for(int i = 0; i < QTD_THREADS_CONSUMIDORAS; i++){
        consumidor_leu[i] = false; //Inicializando todos os consumidores como não tendo lido a última msg
        //O mutex não é necessário ainda já que nenhuma thread produtora vai mexer com essas variáveis e as threads
        //Consumidoras ainda não foram inicializadas
    }

    for(int i = 0; i < QTD_THREADS_CONSUMIDORAS; i++){
        std::cout << "Thread consumidora de id " << i << " iniciando\n";
        thread consumidora(consumer, i); //Iniciando as threads consumidoras
        threads_consumidoras.emplace_back(std::move(consumidora)); //Colocando no vector de threads consumidoras
    }

    for(auto &thread_prod: threads_produtoras){
        thread_prod.join(); //Join nas threads produtoras
    }

    for(auto &thread_cons : threads_consumidoras){
        thread_cons.join(); //Join nas threads consumidoras
    }
}

```

Código:

```

#include <iostream>
#include <thread>
#include <string>
#include <vector>
#include <bits/stdc++.h>

#include <semaphore.h>
#include <unistd.h>

using namespace std;

const int QTD_THREADS_CONSUMIDORAS = 4; //Quantidade de threads consumidoras
const int QTD_THREADS_PRODUTORAS = 4; //Quantidade de threads produtoras

const vector<string> frases{"Arsenal Liverpool Manchester City", "Aston Villa Tottenham
Manchester United",
                            "West Ham Brighton Wolves", "Newcastle Chelsea Fulham Bournemouth",
                            "Crystal Palace Brentford Everton",
                            "Nottingham Forest Luton Town Burnley Sheffield United"};
                            //Frases a serem enviadas (tabela da Premier League quando enviei a primeira atv)

vector<string> buffer; //Vector que serve como um buffer
bool consumidor_leu[QTD_THREADS_CONSUMIDORAS]; //Vetor para indicar qual consumidor
leu a mensagem atual (a primeira) do buffer

```

```

sem_t mutex_buffer; //Mutex para o acesso ao buffer
sem_t mutex_leu; //Mutex para o acesso a variável consumidor_leu

void producer(int i){
    int index_frase = i % frases.size(); //Pega a frase que será mandada pelo produtor
    //Como a quantidade de threads produtoras pode ser maior que a quantidade de frases possíveis,
    //utilizo
    //o mod para garantir que sempre será pego alguma frase válida no vector

    string frase = frases[index_frase]; //Pegar a frase

    stringstream ss(frase); //Inicializando varíavel que fará basicamente o split da mensagem em
    palavras

    string token; //Token que armazenará cada palavra da mensagem

    int contador = 0; //Inicialização do contador para saber se será feito um push bakc no buffer
    //Ou se será apenas alterado a string no buffer

    sem_wait(&mutex_buffer); //Mutex para garantir que o buffer está apenas sendo acessado por
    uma thread
    //Ele é colocado mesmo antes do while para impedir que as mensagens das threadas fiquem
    embaralhadas
    cout << "Thread produtora de id " << i << " enviará mensagem: " << frase << "\n";

    while(ss >> token){ //Enquanto ainda houver palavras na stringstream
        if (contador == 0){ //Se for a primeira palavra
            cout << "Thread produtora de id " << i << " colocou no buffer: " << token << "\n";
            buffer.push_back(token); //Push back no buffer (nova mensagem está sendo inserida)
            contador++; //Indicando que a primeira palavra já foi
            continue;
        }
        cout << "Thread produtora de id " << i << " colocou no buffer: " << token << "\n";

        buffer.at(buffer.size() - 1) = buffer.at(buffer.size() - 1) + ' ' + token; //Pegando a última palavra
        //adicionada (que foi adicionada por essa thread) e adicionada a próxima palavra
        sleep(1);
    }
    sem_post(&mutex_buffer); //Liberando a seção crítica
}

void consumer(int i){
    int id = gettid(); //Pegando o thread id

    while(!buffer.empty()){ //Enquanto ainda existirem palavras a serem lidas
        sem_wait(&mutex_leu); //Entrando na seção crítica da variável consumidor_leu

        bool terminou = true; //Se todos os consumidores leram, terminou
        for(int i = 0; i < QTD_THREADS_CONSUMIDORAS; i++){
            if(consumidor_leu[i] == false){ //Se algum consumidor não leu
                terminou = false; //Não terminou
            }
        }
    }
}

```

```

        break;
    }
}

if(terminou){ //Se terminou
    sem_wait(&mutex_buffer); //Entrando na seção crítica do buffer
    //Ou seja, não dá pra dar esse erase se algum produtor estiver produzindo
    //Ou se algum consumidor estiver consumindo
    buffer.erase(buffer.begin()); //Apaga a primeira mensagem do buffer
    sem_post(&mutex_buffer); //Saindo da seção crítica do buffer

    //Além disso, setando todos os consumidor_leu para falso a fim de liberar os consumidores
    for(int i = 0; i < QTD_THREADS_CONSUMIDORAS; i++){
        consumidor_leu[i] = false;
    }
}
else{
    if(consumidor_leu[i] == true){ //Se o consumidor atual já leu
        sem_post(&mutex_leu); //Libera o mutex da variável consumidor_leu
        continue; //Não vai conseguir ler a mensagem no buffer
    }
}
sem_post(&mutex_leu); //Liberando o mutex de consumidor_leu

sem_wait(&mutex_buffer); //Entrando na seção crítica do buffer
if(!buffer.empty()){ //Se o estiver vazio
    string mensagem = buffer.front(); //Pegando a primeira mensagem do buffer
    cout << "Thread de tid " << id << " leu mensagem: " << mensagem << "\n";

    sem_wait(&mutex_leu); //Entrando na seção crítica de consumidor_leu
    //Isso garante que não vai haver nenhuma thread que está consumindo do buffer alterando
    //a variável consumidor_leu enquanto algum consumidor está checando se todas as threads
    já leram do
    //buffer
    consumidor_leu[i] = true;
    sem_post(&mutex_leu); //Saindo da seção crítica de consumidor_leu

    sleep(1);
}

sem_post(&mutex_buffer); //Saindo da seção crítica do buffer

}
}

int main(void){
    vector<thread> threads_produtoras; //Criação do vector de threads produtoras
    vector<thread> threads_consumidoras; //Criação do vector de threads consumidoras

    //Como os 2 são mutexes, preciso inicializá-los com 1, para que o acesso seja mutualmente
    exclusivo
    sem_init(&mutex_buffer, 0, 1); //Inicialização do mutex do buffer
}

```

```

sem_init(&mutex_leu, 0, 1); //Inicialização do mutex do consumidor_leu

for(int i = 0; i < QTD_THREADS_PRODUTORAS; i++){
    std::cout << "Thread produtora de id " << i << " iniciando\n";
    thread produtora(producer, i); //Início da thread produtora
    threads_produtoras.emplace_back(std::move(produtora)); //

Colocando no vector de threads produtoras
}

for(int i = 0; i < QTD_THREADS_CONSUMIDORAS; i++){
    consumidor_leu[i] = false; //Inicializando todos os consumidores como não tendo lido a última
msg
    //O mutex não é necessário ainda já que nenhuma thread produtora vai mexer com essas
variáveis e as threads
    //Consumidoras ainda não foram inicializadas
}

for(int i = 0; i < QTD_THREADS_CONSUMIDORAS; i++){
    std::cout << "Thread consumidora de id " << i << " iniciando\n";
    thread consumidora(consumer, i); //Iniciando as threads consumidoras
    threads_consumidoras.emplace_back(std::move(consumidora)); //Colocando no vector de
threads consumidoras
}

for(auto &thread_prod: threads_produtoras){
    thread_prod.join(); //Join nas threads produtoras
}

for(auto &thread_cons : threads_consumidoras){
    thread_cons.join(); //Join nas threads consumidoras
}

}

```

1)

```

● (base) victor@victor-IdeaPad-Gaming-3-15IHU6:~/Documentos/P6/Sistemas Operacionais/Tarefa 3 t2$ g++ -o main main.cpp
● (base) victor@victor-IdeaPad-Gaming-3-15IHU6:~/Documentos/P6/Sistemas Operacionais/Tarefa 3 t2$ ./main
Thread produtora de id 0 iniciando
Thread produtora de id 1 iniciando
Thread produtora de id 2 iniciando
Thread produtora de id 3 iniciando
Thread produtora de id 1 enviará mensagem: Aston Villa Tottenham Manchester United Msg 1
Thread produtora de id 1 colocou no buffer: Aston
Thread produtora de id 1 colocou no buffer: Villa
Thread consumidora de id 0 iniciando
Thread consumidora de id 1 iniciando
Thread consumidora de id 2 iniciando
Thread consumidora de id 3 iniciando
Thread produtora de id 1 colocou no buffer: Tottenham
Thread produtora de id 1 colocou no buffer: Manchester
Thread produtora de id 1 colocou no buffer: United
Thread produtora de id 2 enviará mensagem: West Ham Brighton Wolves Msg 2
Thread produtora de id 2 colocou no buffer: West
Thread produtora de id 2 colocou no buffer: Ham
Thread produtora de id 2 colocou no buffer: Brighton
Thread produtora de id 2 colocou no buffer: Wolves
Thread produtora de id 3 enviará mensagem: Newcastle Chelsea Fulham Bournemouth Msg 3
Thread produtora de id 3 colocou no buffer: Newcastle
Thread produtora de id 3 colocou no buffer: Chelsea
Thread produtora de id 3 colocou no buffer: Fulham
Thread produtora de id 3 colocou no buffer: Bournemouth
Thread produtora de id 0 enviará mensagem: Arsenal Liverpool Manchester City Msg 4
Thread produtora de id 0 colocou no buffer: Arsenal
Thread produtora de id 0 colocou no buffer: Liverpool
Thread produtora de id 0 colocou no buffer: Manchester
Thread produtora de id 0 colocou no buffer: City

```

```

Thread produtora de id 0 colocou no buffer: City
Thread de tid 30458 leu mensagem: Aston Villa Tottenham Manchester United
Thread de tid 30459 leu mensagem: Aston Villa Tottenham Manchester United
Thread de tid 30460 leu mensagem: Aston Villa Tottenham Manchester United
Thread de tid 30461 leu mensagem: Aston Villa Tottenham Manchester United
Thread de tid 30460 leu mensagem: West Ham Brighton Wolves
Thread de tid 30458 leu mensagem: West Ham Brighton Wolves
Thread de tid 30459 leu mensagem: West Ham Brighton Wolves Msg 1
Thread de tid 30461 leu mensagem: West Ham Brighton Wolves
Thread de tid 30460 leu mensagem: Newcastle Chelsea Fulham Bournemouth
Thread de tid 30458 leu mensagem: Newcastle Chelsea Fulham Bournemouth
Thread de tid 30459 leu mensagem: Newcastle Chelsea Fulham Bournemouth Msg 2
Thread de tid 30461 leu mensagem: Newcastle Chelsea Fulham Bournemouth
Thread de tid 30460 leu mensagem: Arsenal Liverpool Manchester City
Thread de tid 30458 leu mensagem: Arsenal Liverpool Manchester City
Thread de tid 30460 leu mensagem: Arsenal Liverpool Manchester City Msg 3
Thread de tid 30461 leu mensagem: Arsenal Liverpool Manchester City
Thread de tid 30459 leu mensagem: Arsenal Liverpool Manchester City Msg 4

```

2)

Execução com 16 produtores e 14 consumidores

```
● (base) victor@victor-IdeaPad-Gaming-3-15IHU6:~/Documentos/P6/Sistemas Operacionais/Tarefa 3 t2$ ./main
Thread produtora de id 0 iniciando
Thread produtora de id 1 iniciando
Thread produtora de id 2 iniciando
Thread produtora de id 0 enviará mensagem: Arsenal Liverpool Manchester City Msg 1
Thread produtora de id 0 colocou no buffer: Arsenal
Thread produtora de id 0 colocou no buffer: Liverpool
Thread produtora de id 3 iniciando
Thread produtora de id 4 iniciando
Thread produtora de id 5 iniciando
Thread produtora de id 6 iniciando
Thread produtora de id 7 iniciando
Thread produtora de id 8 iniciando
Thread produtora de id 9 iniciando
Thread produtora de id 10 iniciando
Thread produtora de id 11 iniciando
Thread produtora de id 12 iniciando
Thread produtora de id 13 iniciando
Thread produtora de id 14 iniciando
Thread produtora de id 15 iniciando
Thread consumidora de id 0 iniciando
Thread consumidora de id 1 iniciando
Thread consumidora de id 2 iniciando
Thread consumidora de id 3 iniciando
Thread consumidora de id 4 iniciando
Thread consumidora de id 5 iniciando
Thread consumidora de id 6 iniciando
Thread consumidora de id 7 iniciando
Thread consumidora de id 8 iniciando
Thread consumidora de id 9 iniciando
Thread consumidora de id 10 iniciando
Thread consumidora de id 11 iniciando
Thread consumidora de id 12 iniciando
Thread consumidora de id 13 iniciando
Thread produtora de id 0 colocou no buffer: Manchester
Thread produtora de id 0 colocou no buffer: City
Thread produtora de id 1 enviará mensagem: Aston Villa Tottenham Manchester United Msg 2
Thread produtora de id 1 colocou no buffer: Aston
```

```
Thread produtora de id 1 colocou no buffer: Aston
Thread produtora de id 1 colocou no buffer: Villa
Thread produtora de id 1 colocou no buffer: Tottenham
Thread produtora de id 1 colocou no buffer: Manchester
Thread produtora de id 1 colocou no buffer: United
Thread produtora de id 2 enviará mensagem: West Ham Brighton Wolves Msg 3
Thread produtora de id 2 colocou no buffer: West
Thread produtora de id 2 colocou no buffer: Ham
Thread produtora de id 2 colocou no buffer: Brighton
Thread produtora de id 2 colocou no buffer: Wolves
Thread produtora de id 3 enviará mensagem: Newcastle Chelsea Fulham Bournemouth Msg 4
Thread produtora de id 3 colocou no buffer: Newcastle
Thread produtora de id 3 colocou no buffer: Chelsea
Thread produtora de id 3 colocou no buffer: Fulham
Thread produtora de id 3 colocou no buffer: Bournemouth
Thread produtora de id 4 enviará mensagem: Crystal Palace Brentford Everton Msg 5
Thread produtora de id 4 colocou no buffer: Crystal
Thread produtora de id 4 colocou no buffer: Palace
Thread produtora de id 4 colocou no buffer: Brentford
Thread produtora de id 4 colocou no buffer: Everton
Thread produtora de id 5 enviará mensagem: Nottingham Forest Luton Town Burnley Sheffield United Msg 6
Thread produtora de id 5 colocou no buffer: Nottingham
Thread produtora de id 5 colocou no buffer: Forest
Thread produtora de id 5 colocou no buffer: Luton
Thread produtora de id 5 colocou no buffer: Town
Thread produtora de id 5 colocou no buffer: Burnley
Thread produtora de id 5 colocou no buffer: Sheffield
Thread produtora de id 5 colocou no buffer: United
Thread produtora de id 6 enviará mensagem: Arsenal Liverpool Manchester City Msg 7
Thread produtora de id 6 colocou no buffer: Arsenal
Thread produtora de id 6 colocou no buffer: Liverpool
Thread produtora de id 6 colocou no buffer: Manchester
Thread produtora de id 6 colocou no buffer: City
Thread produtora de id 7 enviará mensagem: Aston Villa Tottenham Manchester United Msg 8
Thread produtora de id 7 colocou no buffer: Aston
Thread produtora de id 7 colocou no buffer: Villa
Thread produtora de id 7 colocou no buffer: Tottenham
Thread produtora de id 7 colocou no buffer: Manchester
```

```

Thread produtora de id 7 colocou no buffer: United
Thread produtora de id 8 enviará mensagem: West Ham Brighton Wolves Msg 9
Thread produtora de id 8 colocou no buffer: West
Thread produtora de id 8 colocou no buffer: Ham
Thread produtora de id 8 colocou no buffer: Brighton
Thread produtora de id 8 colocou no buffer: Wolves
Thread produtora de id 9 enviará mensagem: Newcastle Chelsea Fulham Bournemouth Msg 10
Thread produtora de id 9 colocou no buffer: Newcastle
Thread produtora de id 9 colocou no buffer: Chelsea
Thread produtora de id 9 colocou no buffer: Fulham
Thread produtora de id 9 colocou no buffer: Bournemouth
Thread produtora de id 10 enviará mensagem: Crystal Palace Brentford Everton Msg 11
Thread produtora de id 10 colocou no buffer: Crystal
Thread produtora de id 10 colocou no buffer: Palace
Thread produtora de id 10 colocou no buffer: Brentford
Thread produtora de id 10 colocou no buffer: Everton
Thread produtora de id 11 enviará mensagem: Nottingham Forest Luton Town Burnley Sheffield United Msg 12
Thread produtora de id 11 colocou no buffer: Nottingham
Thread produtora de id 11 colocou no buffer: Forest
Thread produtora de id 11 colocou no buffer: Luton
Thread produtora de id 11 colocou no buffer: Town
Thread produtora de id 11 colocou no buffer: Burnley
Thread produtora de id 11 colocou no buffer: Sheffield
Thread produtora de id 11 colocou no buffer: United
Thread produtora de id 12 enviará mensagem: Arsenal Liverpool Manchester City Msg 13
Thread produtora de id 12 colocou no buffer: Arsenal
Thread produtora de id 12 colocou no buffer: Liverpool
Thread produtora de id 12 colocou no buffer: Manchester
Thread produtora de id 12 colocou no buffer: City
Thread produtora de id 13 enviará mensagem: Aston Villa Tottenham Manchester United Msg 14
Thread produtora de id 13 colocou no buffer: Aston
Thread produtora de id 13 colocou no buffer: Villa
Thread produtora de id 13 colocou no buffer: Tottenham
Thread produtora de id 13 colocou no buffer: Manchester
Thread produtora de id 13 colocou no buffer: United
Thread produtora de id 14 enviará mensagem: West Ham Brighton Wolves Msg 15
Thread produtora de id 14 colocou no buffer: West
Thread produtora de id 14 colocou no buffer: Ham

```

```

Thread de tid 31434 leu mensagem: West Ham Brighton Wolves
Thread de tid 31441 leu mensagem: West Ham Brighton Wolves
Thread de tid 31439 leu mensagem: West Ham Brighton Wolves
Thread de tid 31436 leu mensagem: West Ham Brighton Wolves
Thread de tid 31438 leu mensagem: West Ham Brighton Wolves
Thread de tid 31437 leu mensagem: West Ham Brighton Wolves
Thread de tid 31443 leu mensagem: West Ham Brighton Wolves
Thread de tid 31442 leu mensagem: West Ham Brighton Wolves
Thread de tid 31435 leu mensagem: West Ham Brighton Wolves
Thread de tid 31445 leu mensagem: West Ham Brighton Wolves
Thread de tid 31440 leu mensagem: West Ham Brighton Wolves
Thread de tid 31446 leu mensagem: West Ham Brighton Wolves
Thread de tid 31433 leu mensagem: West Ham Brighton Wolves
Thread de tid 31440 leu mensagem: Newcastle Chelsea Fulham Bournemouth
Thread de tid 31445 leu mensagem: Newcastle Chelsea Fulham Bournemouth
Thread de tid 31436 leu mensagem: Newcastle Chelsea Fulham Bournemouth
Thread de tid 31438 leu mensagem: Newcastle Chelsea Fulham Bournemouth
Thread de tid 31441 leu mensagem: Newcastle Chelsea Fulham Bournemouth
Thread de tid 31442 leu mensagem: Newcastle Chelsea Fulham Bournemouth
Thread de tid 31434 leu mensagem: Newcastle Chelsea Fulham Bournemouth
Thread de tid 31444 leu mensagem: Newcastle Chelsea Fulham Bournemouth
Thread de tid 31446 leu mensagem: Newcastle Chelsea Fulham Bournemouth
Thread de tid 31437 leu mensagem: Newcastle Chelsea Fulham Bournemouth
Thread de tid 31435 leu mensagem: Newcastle Chelsea Fulham Bournemouth
Thread de tid 31443 leu mensagem: Newcastle Chelsea Fulham Bournemouth
Thread de tid 31439 leu mensagem: Newcastle Chelsea Fulham Bournemouth
Thread de tid 31433 leu mensagem: Newcastle Chelsea Fulham Bournemouth
Thread de tid 31442 leu mensagem: Crystal Palace Brentford Everton
Thread de tid 31436 leu mensagem: Crystal Palace Brentford Everton
Thread de tid 31444 leu mensagem: Crystal Palace Brentford Everton
Thread de tid 31434 leu mensagem: Crystal Palace Brentford Everton
Thread de tid 31439 leu mensagem: Crystal Palace Brentford Everton
Thread de tid 31443 leu mensagem: Crystal Palace Brentford Everton
Thread de tid 31445 leu mensagem: Crystal Palace Brentford Everton
Thread de tid 31440 leu mensagem: Crystal Palace Brentford Everton
Thread de tid 31435 leu mensagem: Crystal Palace Brentford Everton
Thread de tid 31437 leu mensagem: Crystal Palace Brentford Everton
Thread de tid 31441 leu mensagem: Crystal Palace Brentford Everton
Thread de tid 31444 leu mensagem: West Ham Brighton Wolves
Thread de tid 31434 leu mensagem: West Ham Brighton Wolves

```

Msg 3

Msg 4

Msg 5


```
Thread de tid 31430 leu mensagem: ASTON VILLA - TOTTENHAM MANCHESTER UNITED  
Thread de tid 31434 leu mensagem: West Ham Brighton Wolves  
Thread de tid 31445 leu mensagem: West Ham Brighton Wolves  
Thread de tid 31438 leu mensagem: West Ham Brighton Wolves  
Thread de tid 31435 leu mensagem: West Ham Brighton Wolves  
Thread de tid 31439 leu mensagem: West Ham Brighton Wolves  
Thread de tid 31446 leu mensagem: West Ham Brighton Wolves  
Thread de tid 31444 leu mensagem: West Ham Brighton Wolves  
Thread de tid 31433 leu mensagem: West Ham Brighton Wolves  
Thread de tid 31437 leu mensagem: West Ham Brighton Wolves  
Thread de tid 31443 leu mensagem: West Ham Brighton Wolves  
Thread de tid 31440 leu mensagem: West Ham Brighton Wolves  
Thread de tid 31441 leu mensagem: West Ham Brighton Wolves  
Thread de tid 31442 leu mensagem: West Ham Brighton Wolves  
Thread de tid 31436 leu mensagem: West Ham Brighton Wolves  
Thread de tid 31440 leu mensagem: Newcastle Chelsea Fulham Bournemouth  
Thread de tid 31443 leu mensagem: Newcastle Chelsea Fulham Bournemouth  
Thread de tid 31433 leu mensagem: Newcastle Chelsea Fulham Bournemouth  
Thread de tid 31435 leu mensagem: Newcastle Chelsea Fulham Bournemouth  
Thread de tid 31446 leu mensagem: Newcastle Chelsea Fulham Bournemouth  
Thread de tid 31439 leu mensagem: Newcastle Chelsea Fulham Bournemouth  
Thread de tid 31438 leu mensagem: Newcastle Chelsea Fulham Bournemouth  
Thread de tid 31434 leu mensagem: Newcastle Chelsea Fulham Bournemouth  
Thread de tid 31444 leu mensagem: Newcastle Chelsea Fulham Bournemouth  
Thread de tid 31442 leu mensagem: Newcastle Chelsea Fulham Bournemouth  
Thread de tid 31445 leu mensagem: Newcastle Chelsea Fulham Bournemouth  
Thread de tid 31437 leu mensagem: Newcastle Chelsea Fulham Bournemouth  
Thread de tid 31441 leu mensagem: Newcastle Chelsea Fulham Bournemouth  
Thread de tid 31436 leu mensagem: Newcastle Chelsea Fulham Bournemouth
```

Msg 15

Msg 16