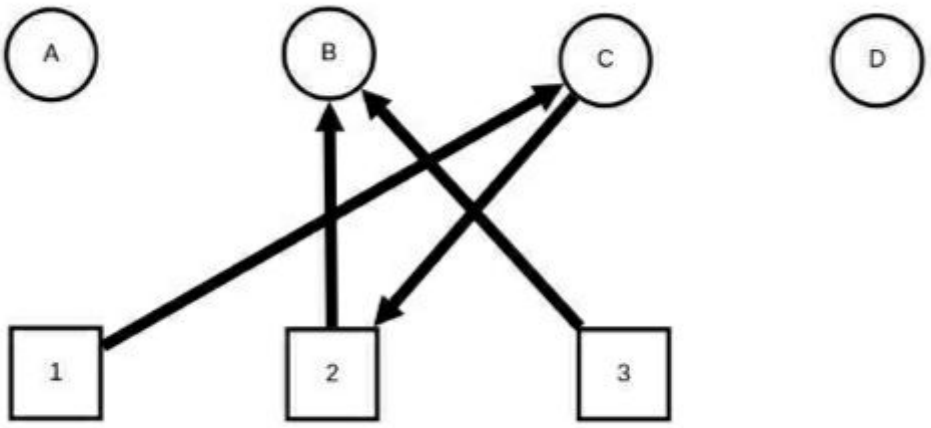
Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação
Programação Concorrente

Relatório de Entrega de Atividades

Aluno(s): Amanda Oliveira Alves e Fillype Alves do Nascimento
Matrícula: 15/0116276 e 16/0070431
Atividade: Aula Prática 06 - Deadlocks

1.1.1 - Estado final do diagrama após o último evento citado.



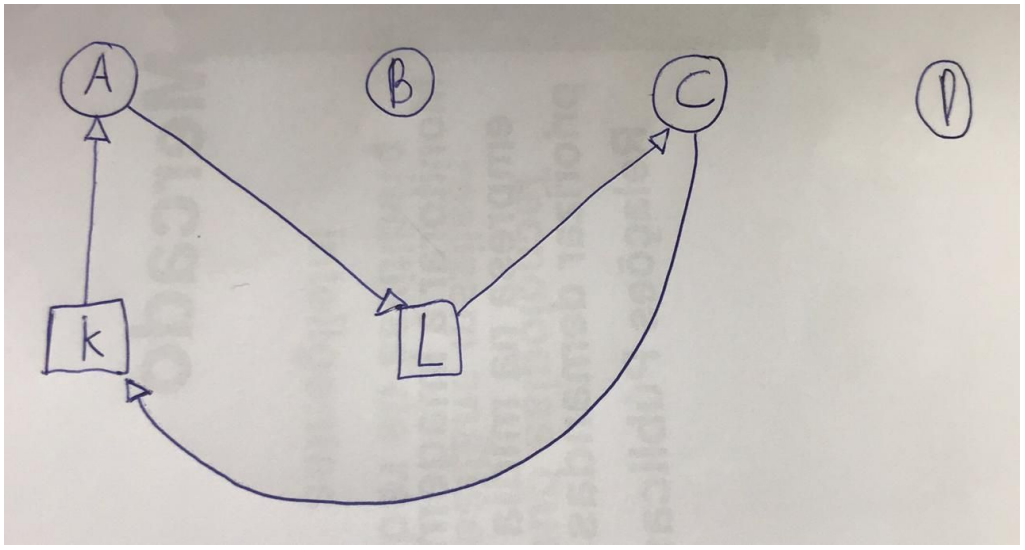
1.1.2 -

A requisita K

C requisita L

A requisita L

C requisita K



2.1.1

```
// autor: Amanda Oliveira Alves e Fillype Alves do Nascimento
// arquivo: 2.1.1.c
// atividade: 2.1.1

#include <stdio.h>
#include <pthread.h>
#include <unistd.h>
#include <stdlib.h>

#define TOTAL 5
#define DIREITA (id_filosofo + 1) % TOTAL
#define ESQUERDA (id_filosofo + TOTAL - 1) % TOTAL
#define PENSANDO 0
#define COMFOME 1
#define COMENDO 2

int estado[TOTAL];
pthread_t jantar[TOTAL];
pthread_mutex_t mutex;
pthread_mutex_t mux_filo [TOTAL];

void *filosofo(void *param);
```

```
void pegar_garfo(int id_filosofo);
void devolver_garfo(int id_filosofo);
void intencao(int id_filosofo);
void comer(int id_filosofo);
void pensar(int id_filosofo);

void *filosofo(void *vparam){

   int *id = (int *)(vparam);

   printf("Filósofo %d foi criado com sucesso\n", *(id));

   while(1){

      pensar(*(id));
      pegar_garfo(*(id));
      comer(*(id));
      devolver_garfo(*(id));
   }

   pthread_exit((void*)0);
}

void pegar_garfo(int id_filosofo){

   pthread_mutex_lock(&(mutex));
   printf("Filósofo %d está com fome\n", id_filosofo);
   estado[id_filosofo] = COMFOME;
   intencao(id_filosofo);
   pthread_mutex_unlock(&(mutex));
   pthread_mutex_lock(&(mux_filo[id_filosofo]));
}

void devolver_garfo(int id_filosofo){

   pthread_mutex_lock(&(mutex));
   printf("Filósofo %d está pensando\n", id_filosofo);
```

```
   estado[id_filosofo] = PENSANDO;
   intencao(ESQUERDA);
   intencao(DIREITA);
   pthread_mutex_unlock(&(mutex));
}

void intencao(int id_filosofo){

   printf("Filósofo %d está com intenção de comer\n", id_filosofo);
   if((estado[id_filosofo] == COMFOME) && (estado[ESQUERDA] != COMENDO) &&
(estado[DIREITA] != COMENDO)){

      printf("Filósofo %d ganhou a vez de comer\n", id_filosofo);
      estado[id_filosofo] = COMENDO;
      pthread_mutex_unlock(&(mux_filo[id_filosofo]));
   }
}

void pensar(int id_filosofo){

   printf("Filósofo %d está pensando\n", id_filosofo);
   sleep(1);
}

void comer(int id_filosofo){
   printf("Filósofo %d está comendo\n", id_filosofo);
   sleep(rand() % 5);
}

int main(){

   int i;

   pthread_mutex_init( &(mutex), NULL);
   for(i= 0;i< TOTAL;i++){
      pthread_mutex_init(&(mux_filo[i]), NULL);
   }
```

```
   for(i=0;i< TOTAL;i++){
      pthread_create(&(jantar[i]), NULL, filosofo, (void *)&(i));
   }

   pthread_mutex_destroy(&(mutex));
   for(i=0;i<TOTAL;i++){
      pthread_mutex_destroy(&(mux_filo[i]));
   }
   pthread_exit(NULL);

   return 0;
}
```

3.1.1

```
// autor: Amanda Oliveira Alves e Fillype Alves do Nascimento
// arquivo: 3.1.1.c
// atividade: 3.1.1

#include <pthread.h>
#include <stdio.h>
#include <stdlib.h>

int var = 0;
pthread_mutex_t posse;

void* contador ( void* arg ) {
    int *i = (int *) arg;

    pthread_mutex_lock(&posse);
```

```
    var++;
    // printf("Thread %d incrementou. Novo Valor: %d.\n", *i, var);

    pthread_mutex_unlock(&posse);
    pthread_exit(NULL);
}

int main (void) {
    pthread_t t[10];

    pthread_mutex_init(&posse, 0);

    for (int i = 0; i < 10; i++) {
        int *id;
        id = malloc(sizeof(int));
        *id = i;
        pthread_create(&t[i], NULL, contador, (int *) id);
    }

    pthread_mutex_lock(&posse);
    var++; // A main também irá incrementar
    // printf("Main incrementou. Novo Valor: %d.\n", var);
    pthread_mutex_unlock(&posse);

    for (int i = 0; i < 10; i++) {
        pthread_join(t[i], NULL);
    }

    return 0;
}
```

```
/*
Valgrind

==20901== Memcheck, a memory error detector
==20901== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==20901== Using Valgrind-3.15.0 and LibVEX; rerun with -h for copyright
info
==20901== Command: ./3
==20901==
==20901==
==20901== HEAP SUMMARY:
==20901==     in use at exit: 0 bytes in 0 blocks
==20901==   total heap usage: 10 allocs, 10 frees, 2,720 bytes allocated
==20901==
==20901== All heap blocks were freed -- no leaks are possible
==20901==
==20901== For lists of detected and suppressed errors, rerun with: -s
==20901== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)

Helgrind

==21012== Helgrind, a thread error detector
==21012== Copyright (C) 2007-2017, and GNU GPL'd, by OpenWorks LLP et al.
==21012== Using Valgrind-3.15.0 and LibVEX; rerun with -h for copyright
info
==21012== Command: ./3
==21012==
==21012== ---Thread-Announcement------------------------------------------
==21012==
==21012== Thread #3 was created
==21012==    at 0x4999542: clone (in /usr/lib64/libc-2.29.so)
==21012==    by 0x488339E: create_thread (in
/usr/lib64/libpthread-2.29.so)
==21012==    by 0x4884D64: pthread_create@@GLIBC_2.2.5 (in
/usr/lib64/libpthread-2.29.so)
==21012==    by 0x483F575: pthread_create_WRK (hg_intercepts.c:427)
==21012==    by 0x4840678: pthread_create@* (hg_intercepts.c:460)
```

```
==21012==    by 0x401188: main (3.1.1.c:13)
==21012==
==21012== ---Thread-Announcement------------------------------------------
==21012==
==21012== Thread #2 was created
==21012==    at 0x4999542: clone (in /usr/lib64/libc-2.29.so)
==21012==    by 0x488339E: create_thread (in
/usr/lib64/libpthread-2.29.so)
==21012==    by 0x4884D64: pthread_create@@GLIBC_2.2.5 (in
/usr/lib64/libpthread-2.29.so)
==21012==    by 0x483F575: pthread_create_WRK (hg_intercepts.c:427)
==21012==    by 0x4840678: pthread_create@* (hg_intercepts.c:460)
==21012==    by 0x401188: main (3.1.1.c:13)
==21012==
==21012== ----------------------------------------------------------------
==21012==
==21012== Possible data race during read of size 4 at 0x404030 by thread
#3
==21012== Locks held: none
==21012==    at 0x40113E: contador (3.1.1.c:6)
==21012==    by 0x483F76D: mythread_wrapper (hg_intercepts.c:389)
==21012==    by 0x48844BF: start_thread (in /usr/lib64/libpthread-2.29.so)
==21012==    by 0x4999552: clone (in /usr/lib64/libc-2.29.so)
==21012==
==21012== This conflicts with a previous write of size 4 by thread #2
==21012== Locks held: none
==21012==    at 0x401147: contador (3.1.1.c:6)
==21012==    by 0x483F76D: mythread_wrapper (hg_intercepts.c:389)
==21012==    by 0x48844BF: start_thread (in /usr/lib64/libpthread-2.29.so)
==21012==    by 0x4999552: clone (in /usr/lib64/libc-2.29.so)
==21012==  Address 0x404030 is 0 bytes inside data symbol "var"
==21012==
==21012== ----------------------------------------------------------------
==21012==
==21012== Possible data race during write of size 4 at 0x404030 by thread
#3
==21012== Locks held: none
```

```
==21012==    at 0x401147: contador (3.1.1.c:6)
==21012==    by 0x483F76D: mythread_wrapper (hg_intercepts.c:389)
==21012==    by 0x48844BF: start_thread (in /usr/lib64/libpthread-2.29.so)
==21012==    by 0x4999552: clone (in /usr/lib64/libc-2.29.so)
==21012==
==21012== This conflicts with a previous write of size 4 by thread #2
==21012== Locks held: none
==21012==    at 0x401147: contador (3.1.1.c:6)
==21012==    by 0x483F76D: mythread_wrapper (hg_intercepts.c:389)
==21012==    by 0x48844BF: start_thread (in /usr/lib64/libpthread-2.29.so)
==21012==    by 0x4999552: clone (in /usr/lib64/libc-2.29.so)
==21012==  Address 0x404030 is 0 bytes inside data symbol "var"
==21012==
==21012== ---Thread-Announcement------------------------------------------
==21012==
==21012== Thread #1 is the program's root thread
==21012==
==21012== ---Thread-Announcement------------------------------------------
==21012==
==21012== Thread #11 was created
==21012==    at 0x4999542: clone (in /usr/lib64/libc-2.29.so)
==21012==    by 0x488339E: create_thread (in
/usr/lib64/libpthread-2.29.so)
==21012==    by 0x4884D64: pthread_create@@GLIBC_2.2.5 (in
/usr/lib64/libpthread-2.29.so)
==21012==    by 0x483F575: pthread_create_WRK (hg_intercepts.c:427)
==21012==    by 0x4840678: pthread_create@* (hg_intercepts.c:460)
==21012==    by 0x401188: main (3.1.1.c:13)
==21012==
==21012== ----------------------------------------------------------------
==21012==
==21012== Possible data race during read of size 4 at 0x404030 by thread
#1
==21012== Locks held: none
==21012==    at 0x401193: main (3.1.1.c:16)
==21012==
==21012== This conflicts with a previous write of size 4 by thread #11
```

```
==21012== Locks held: none
==21012==    at 0x401147: contador (3.1.1.c:6)
==21012==    by 0x483F76D: mythread_wrapper (hg_intercepts.c:389)
==21012==    by 0x48844BF: start_thread (in /usr/lib64/libpthread-2.29.so)
==21012==    by 0x4999552: clone (in /usr/lib64/libc-2.29.so)
==21012==  Address 0x404030 is 0 bytes inside data symbol "var"
==21012==
==21012== ----------------------------------------------------------------
==21012==
==21012== Possible data race during write of size 4 at 0x404030 by thread
#1
==21012== Locks held: none
==21012==    at 0x40119C: main (3.1.1.c:16)
==21012==
==21012== This conflicts with a previous write of size 4 by thread #11
==21012== Locks held: none
==21012==    at 0x401147: contador (3.1.1.c:6)
==21012==    by 0x483F76D: mythread_wrapper (hg_intercepts.c:389)
==21012==    by 0x48844BF: start_thread (in /usr/lib64/libpthread-2.29.so)
==21012==    by 0x4999552: clone (in /usr/lib64/libc-2.29.so)
==21012==  Address 0x404030 is 0 bytes inside data symbol "var"
==21012==
==21012==
==21012== Use --history-level=approx or =none to gain increased speed, at
==21012== the cost of reduced accuracy of conflicting-access information
==21012== For lists of detected and suppressed errors, rerun with: -s
==21012== ERROR SUMMARY: 20 errors from 4 contexts (suppressed: 0 from 0)
```

HELGRIND APÓS ALTERAÇÕES NO CÓDIGO

```
==23229== Helgrind, a thread error detector
==23229== Copyright (C) 2007-2017, and GNU GPL'd, by OpenWorks LLP et al.
==23229== Using Valgrind-3.15.0 and LibVEX; rerun with -h for copyright
info
==23229== Command: ./3
==23229==
==23229==
```

```
==23229== Use --history-level=approx or =none to gain increased speed, at
==23229== the cost of reduced accuracy of conflicting-access information
==23229== For lists of detected and suppressed errors, rerun with: -s
==23229== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 491 from
28)
*/
```