15-213/5A3/6gh3neFitrParb Extalina ReWielyw

https://powcoder.com

Kashish & Ishita Add We Chat powcoder

# Final Exam Logistics

- Cheat sheets 2 double sided 8.5 x 11 in.
- Join Zoom, turn on video AND microphone. Show ID and cheatsheet to TA on video.
- You will receive tank expailworth the tink to the zoom call, the exam time, and more detailed logistics soon. Add WeChat powcoder
- 8 categories of questions:
  - Malloc, VM, Processes, Signals, IO, Threads, ThreadSync, Multiple Choice (pre-midterm)



Virtual Address - 18 Bits

Physical Address -A12 Bits nment

Page Size - 512 Bytes

https://p

TLB is 8-way set associative

Add We

Cache is 2-way set associative

Final S-02 (#5)

Lecture 18: VM - Systems

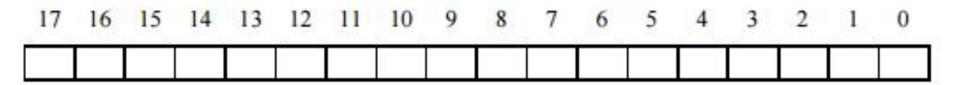
		Page	Table		
VPN	PPN	Valid	VPN	PPN	Valid
000	7	0	010	1	0
001	5	0	011	3	0
002	1	1	012	3	0
003	5	0	013	0	0
004	0	0	014	6	1
Opp	iec	t E	X <sup>0</sup> Þfr	5	<b>6</b> 0
006	2	0	016	7	0
007	4	1	017	2	1
008	20	ler.	$C_0^{018}$	0	0
009	cyc	ict.	019	2	0
00A	3	0	01A	1	0
00B	0	0	01B	3	0
000	at t	OW	700	der	0
00D	3	0	01D	7	0
00E	4	0	01E	5	1
00F	7	1	01F	0	0

TLB												
Index	Tag	PPN	Valid									
0	55	6	0									
	48	F	1									
	00	A	0									
	32	9	1									
	6A	3	1									
	56	1	0									
	60	4	1									
	78	9	0									
1	71	5	1									
	31	A	1									
	53	F	0									
	87	8	0									
	51	D	0									
	39	E	1									
	43	В	0									
· 4	73	2	1									

	2-way Set Associative Cache														
Index	Tag	Valid	Byte 0	Byte 1	Byte 2	Byte 3	Tag	Valid	Byte 0	Byte 1	Byte 2	Byte 3			
0	7A	1	09	EE	12	64	00	0	99	04	03	48			
1	02	0	60	17	18	19	7F	1	FF	BC	0B	37			
2	55	1	30	EB	C2	0D	0B	0	8F	E2	05	BD			
3	07	1	03	04	05	06	5D	1	7A	08	03	22			

#### Label the following:

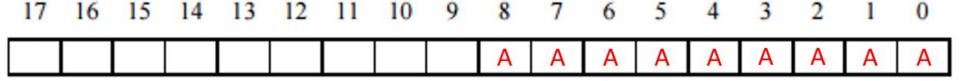
- VPO: Virtual Page Offset VPN: Virtual Page Number Exam Help
- TLBI: TLB Indexttps://powcoder.com
- TLBT: TLB Tag



Label the following:

(A) VPO: Virtual Page Offset - Location in the page Page Size = 512 Bytes = 29 - Need 9 bits

https://powcoder.com



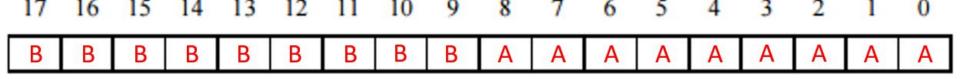
Label the following:

- VPO: Virtual Page Offset VPN: Virtual Page Number Everything Else

https://powcoder.com

#### Label the following:

- VPO: Virtual Page Offset VPN: Virtual Page Number Exam Help
- TLBI: TLB Indextrosogation in the Tork Cache



Label the following:

- VPO: Virtual Page Offset VPN: Virtual Page Number Exam Help
- TLBI: TLB Indextrosociation in the Toris Cache 2 Indices  $\rightarrow$  1 Bit

Add WeChat powcoder

TLBI

#### Label the following:

- VPO: Virtual Page Offset VPN: Virtual Page Number Exam Help
- TLBI: TLB Indexttps://powcoder.com
- TLBT: TLB Tag Everything Else

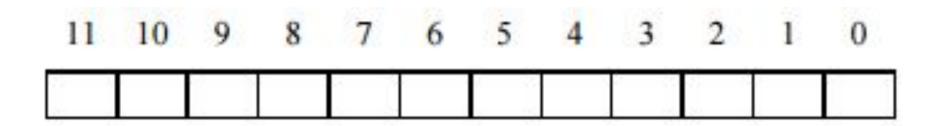
Add WeChat powcoder



**TLBT** TLBI

#### Label the following:

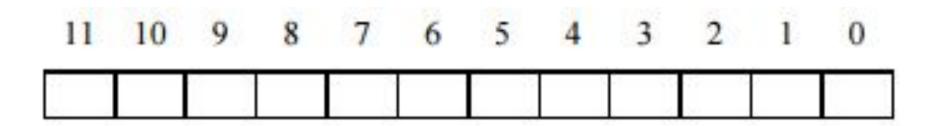
- (A) PPO: Physical Page Offset.
  (B) PPN: Physical Page Number
- (C) CO: Cache Offaetps://powcoder.com
- CI: Cache Index
- (E) CT: Cache TagAdd WeChat powcoder



Label the following:

(A) PPO: Physical Page Offset. Assignment Project Exam Help

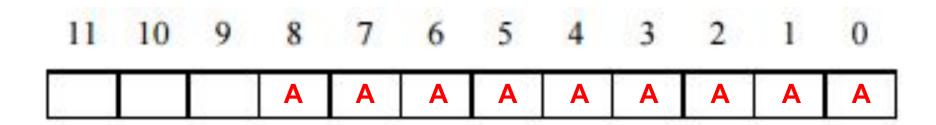
https://powcoder.com



Label the following:

(A) PPO: Physical Page Offset - Same as VPO Assignment Project Exam Help

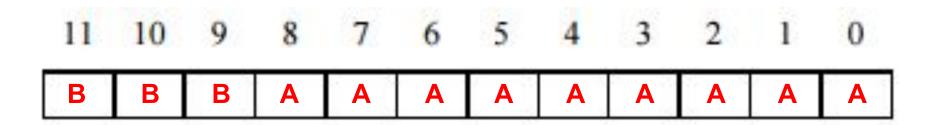
https://powcoder.com



Label the following:

- PPO: Physical Page Offset. Same as VPO Assignment Project Exam Help PPN: Physical Page Number Everything Else

https://powcoder.com



#### Label the following:

- PPO: Physical Page Offset. Same as VPO Assignment Project Exam Help PPN: Physical Page Number Everything Else
- CO: Cache Offset ps. Offset in alackom



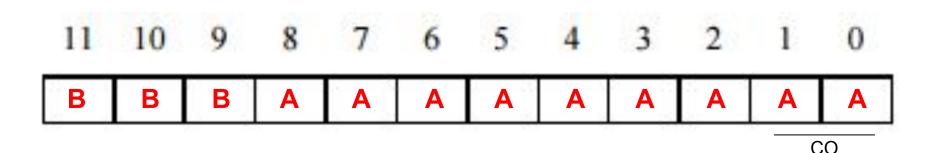
Label the following:

- PPO: Physical Page Offset. Same as VPO Assignment Project Exam Help PPN: Physical Page Number Everything Else
- CO: Cache Offset in alockom 4 Byte Blocks → 2 Bits Add WeChat powcoder



#### Label the following:

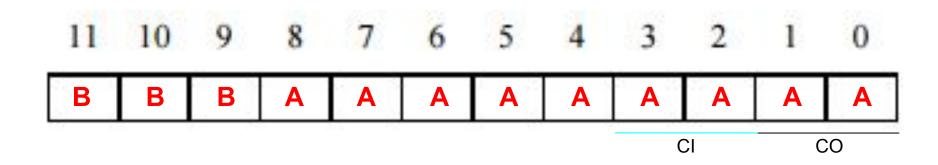
- PPO: Physical Page Offset. Same as VPO Assignment Project Exam Help PPN: Physical Page Number Everything Else
- (C) CO: Cache Offset in allockom
- CI: Cache Index



#### Label the following:

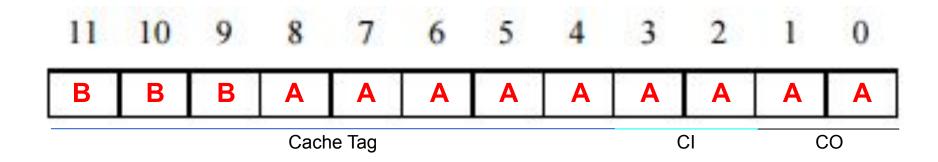
- PPO: Physical Page Offset. Same as VPO Assignment Project Exam Help PPN: Physical Page Number Everything Else
- (C) CO: Cache Offset in allockom
- CI: Cache Index

AddhanceShat 20 Butscoder



#### Label the following:

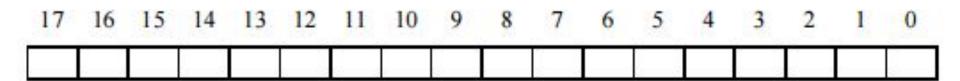
- PPO: Physical Page Offset. Same as VPO Assignment Project Exam Help PPN: Physical Page Number Everything Else
- CO: Cache Offset ps. Offset in allockom
- CI: Cache Index
- CT: Cache TagAdfoleNythichgatEpoewcoder



Now to the actual question!

Q) Translate the following address: 0x1A9F4
Assignment Project Exam Help

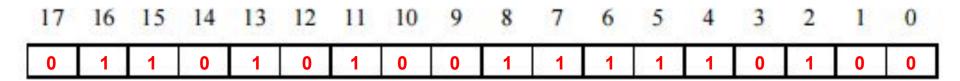
https://powcoder.com



Now to the actual question!

- Q) Translate the following address: 0x1A9F4

  1. Write down bit representation

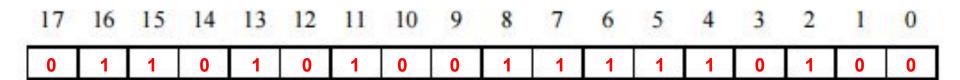


Now to the actual question!

- Q) Translate the following address: 0x1A9F4

  1. Write down bit representation
- Extract Information:

VPN: 0xttps://powcoder.com BT: 0x??

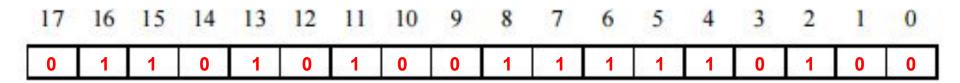


Now to the actual question!

- Q) Translate the following address: 0x1A9F4

  1. Write down bit representation
- Extract Information:

VPN: 0xttps://powcoder.com BT: 0x??

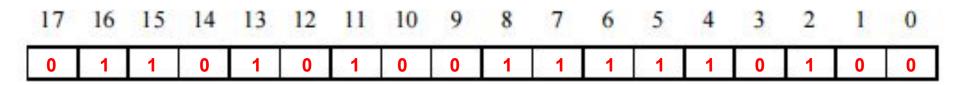


Now to the actual question!

- Q) Translate the following address: 0x1A9F4

  1. Write down bit representation
- Extract Information:

VPN: 0xttps://powcoder.com BT: 0x??



Now to the actual question!

- Q) Translate the following address: 0x1A9F4

  1. Write down bit representation
- Extract Information:

VPN: 0xHttps://posycoder.com BT: 0x6A

Index	Tag	PPN	Valid
0	55	6	0
	48	F	1
	00	A	0
	32	9	1
	6A	3	1
	56	1	0
	60	4	1
	78	9	0
1	71	5	1
	31	A	1
	53	F	0
	87	8	0
	51	D	0
	39	E	1
	43	В	0
	73	2	1

i							4				1		1			1	_	
	1/	10	15	14	13	12	11	10	,	0	,	O	,	*	,	-	1	U

	0
--	---

Now to the actual question!

- Q) Translate the following address: 0x1A9F4

  1. Write down bit representation
- 2. Extract Information:

VPN: 0xftps://posycoder.comest: 0x6A

TLB												
Index	Tag	PPN	Valid									
0	55	6	0									
	48	F	1									
	00	A	0									
	32	9	1									
	6A	3	1									
	56	1	0									
	60	4	1									
	78	9	0									
1	71	5	1									
	31	A	1									
	53	F	0									
	87	8	0									
	51	D	0									
	39	E	1									
	43	В	0									
	73	2	1									

1,	10	13	14	13	12	1.1	10		0	,	O	1	1	3	-	1	U
0	1	1	0	1	0	1	0	0	1	1	1	1	1	0	1	0	0

Now to the actual question!

- Q) Translate the following address: 0x1A9F4

  1. Write down bit representation
- Extract Information:

W	TI	LB	
Index	Tag	PPN	Valid
0	55	6	0
	48	F	1
	00	A	0
	32	9	1
	6A	3	1
	56	1	0
	60	4	1
	78	9	0
1	71	5	1
	31	A	1
	53	F	0
	87	8	0
	51	D	0
	39	E	1
	43	В	0
	73	2	1

0.00	SER.	12000	4.8	H.GYA	17.7		435	3/200	70		. "	2798	180	-03	2000	4	1,000
0	1	1	0	_1	0	. 1	0	0	1	1	1	1	1	0	1	0	0

Now to the actual question!

- Q) Translate the following address: 0x1A9F4

  1. Write down bit representation
- 2. Extract Information:

VPN: 0xHttps://popy.coder.com BT: 0x6A

TLB												
Index	Tag	PPN	Valid									
0	55	6	0									
	48	F	1									
	00	A	0									
	32	9	1									
	6A	3	1									
	56	1	0									
	60	4	1									
	78	9	0									
1	71	5	1									
	31	A	1									
	53	F	0									
	87	8	0									
	51	D	0									
	39	E	1									
	43	В	0									
	73	2	1									

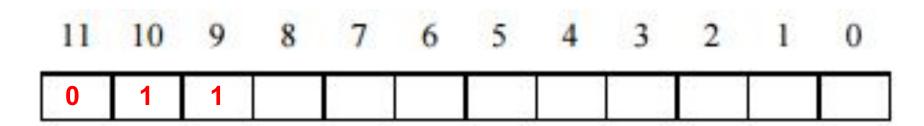
- 10	10			100			10					- To	180	10	<u>,                                    </u>		
0	1	1	0	1	0	1	0	0	1	1	1	1	1	0	1_	0	0

Now to the actual question!

- Q) Translate the following address: 0x1A9F4

  1. Write down bit representation

- Extract Information
   Put it all together: The oxide properties of the content of the c

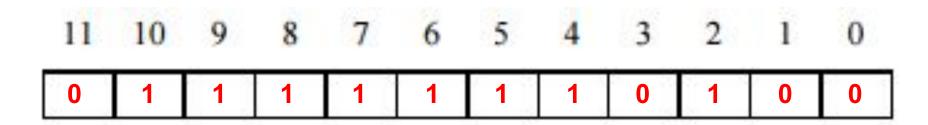


Now to the actual question!

- Q) Translate the following address: 0x1A9F4

  1. Write down bit representation

- 2. Extract Information
  3. Put it all together: The content of t



#### Q) What is the value of the address?

### Assignment Project Exam Help

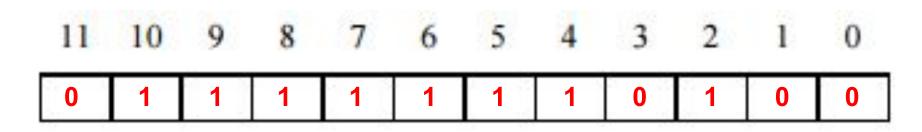
CO: 0x?? CI: 0x?? CT: 0x?? Cache Hit: Y/N? Value:0x?? https://powcoder.com



#### Q) What is the value of the address?

1. Extract more information Assignment Project Exam Help

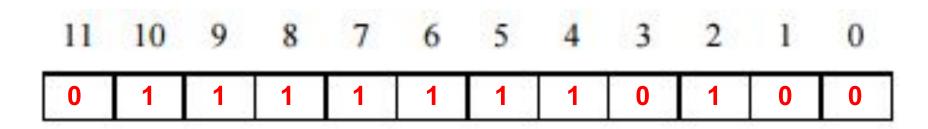
CO: 0x00 CI: 0x?? CT: 0x?? Cache Hit: Y/N? Value:0x?? https://powcoder.com



#### Q) What is the value of the address?

1. Extract more information Assignment Project Exam Help

CO: 0x00 CI: 0x01 CT: 0x?? Cache Hit: Y/N? Value:0x?? https://powcoder.com



#### Q) What is the value of the address?

- 1. Extract more information
- 2. Go to Cache Tablenment Project Exam Help

CO: 0x00 CI: 0x01 CT: 0x7F Cache Hit: Y/N? Value:0x?? https://powcoder.com

	2-way Set Associative Cache											
Index	Tag	Valid	Byte 0	Byte 1	Byte 2	Byte 3	Tag	Valid	Byte 0	Byte 1	Byte 2	Byte 3
0	7A	1	08	EE	<b>V</b> 22	64	20	10 (	197/	CQ4	<b>P</b> 1	48
1	02	0	60	17	18	19	7F	Y	FF	BC	0В	37
2	55	1	30	EB	C2	0D	0B	0	8F	E2	05	BD
3	07	1	03	04	05	06	5D	1	7A	08	03	22

		-	40	_	-	_		-	-		-
11	10	9	×	7	6	. 5	4	3	2	12	()
	4 10		4.3			-			-		- 1

	0	1	1	1	1	1	1	1	0	1	0	0
- 1												

#### Q) What is the value of the address?

- 1. Extract more information
- 2. Go to Cache Assignment Project Exam Help

CO: 0x00 CI: 0x01 CT: 0x7F Cache Hit: Y Value:0x?? https://powcoder.com

	2-way Set Associative Cache											
Index	Tag	Valid	Byte 0	Byte 1	Byte 2	Byte 3	Tag	Valid	Byte 0	Byte 1	Byte 2	Byte 3
0	7A	1	09	EE	<b>V</b> 22	64	20	10 (	197/	C04	(P)	48
1	02	0	60	17	18	19	7F	H,	FF	BC	0B	37
2	55	1	30	EB	C2	0D	0B	0	8F	E2	05	BD
3	07	1	03	04	05	06	5D	1	7A	08	03	22

11	10	9	8	7	6	5	4	3	2	1	0
			-		100			-	_		4.0

	0	1	1	1	1	1	1	1	0	1	0	0
--	---	---	---	---	---	---	---	---	---	---	---	---

#### Q) What is the value of the address?

- 1. Extract more information
- 2. Go to Cache Tablenment Project Exam Help

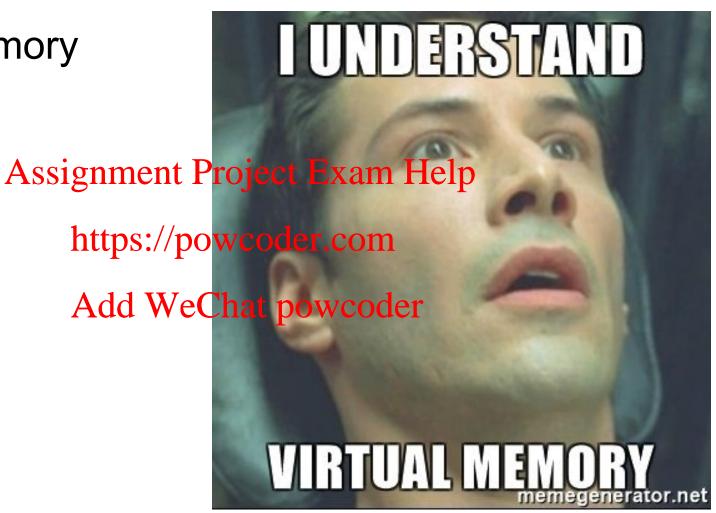
CO: 0x00 CI: 0x01 CT: 0x7F Cache Hit: Y Value:0xFF https://powcoder.com

	2-way Set Associative Cache											
Index	Tag	Valid	Byte 0	Byte 1	Byte 2	Byte 3	Tag	Valid	Byte 0	Byte 1	Byte 2	Byte 3
0	7A	1	08	EE	<b>V</b> /2	64	90	10	1987	C04	(P)	48
1	02	0	60	17	18	19	7F	Y	FF	BC	0B	37
2	55	1	30	EB	C2	0D	0B	0	8F	E2	05	BD
3	07	1	03	04	05	06	5D	1	7A	08	03	22

11	10	9	8	7	6	5	4	3	2	1	0
	10	1	4.4		100	April 1			_		

0 1 1 1 1 1 1 1 1 0	1 0	0
---------------------	-----	---

## **Virtual Memory**



#### Threads



#### **Threads**

Given this code, what variables do you think are shared?

```
#include <stdio.h>
#include <pthread.hAssignment Project ExamiHerpadA(void *vargp) {
                                                            long instance = (long)vargp;
#define NUM THREADS 2
                                                            static int cnt = 0;
int balance = 10;
                         https://powcoder.com
                                                           deposit(4);
                                                           withdraw(11);
                                                           return NULL;
 int main() {
    int i;
    pthread_t tid[NUM_THREADAIdd WeChat powcoder
    pthread create(&tid[0], NULL, threadA, (void*)0);
                                                        void *threadB() {
    pthread create(&tid[1], NULL, threadB, (void*)0);
                                                           withdraw(6);
    for (i = 0; i < NUM THREADS; i++) {
                                                           deposit(3);
        pthread join(tid[i], NULL);
                                                           withdraw(7);
                                                           return NULL;
    printf("balance: %d\n", balance); // What is balance?
    return 0;
```

Which variables can be shared by multiple threads simultaneously in this program?

Assignment Project Exam Help

- (A) i https://powcoder.com
- (B) balance
- (C) instance Add WeChat powcoder
- (D) cnt
- (E) None of the above

Which variables can be shared by multiple threads simultaneously in this program?

Assignment Project Exam Help

- (A) i https://powcoder.com
- (B) balance
- (C) instance Add WeChat powcoder
- (D) cnt
- (E) None of the above

Answer: B

- i is a local variable so it isn't shared.
- Assignment Project Exam Help balance is a global variable so it's shared.

https://powcoder.com
(C) instance is local to threadA() so it isn't shared.

#### Add WeChat powcoder

cnt is a static variable, so it retains its value even outside the scope in which it was defined, so it isn't shared.

```
Given the withdraw() and deposit() functions, what are the possible
outputs? (balance = 10 initially)
int withdraw(int Assignment Projectie * varge) {
                                         long instance = (long)vargp;
    if (balance >= amt) {
                                         static int cnt = 0;
       balance = balance = balance point powcoderecom(4);
       return 0;
                                         withdraw(11);
    } else {
                                         return NULL;
                    Add WeChatpowcoder
       return -1;
                                     void *threadB() {
int deposit(int amt) {
                                         withdraw(6);
     balance = balance + amt;
                                         deposit(3);
     sleep(2);
                                         withdraw(7);
     return 0;
                                         return NULL;
```

What can be the value of balance?

- (A) balance: O Balance
- (B) balance: -3 https://powcoder.com
- (C) balance: 14
- (D) balance: 6 Add WeChat powcoder
- (E) balance: 17
- (F) balance: 4

What can be printed at the indicated line?

- (A) balance: O Balance
- (B) balance: -3 https://powcoder.com
- (C) balance: 14
- (D) balance: 6 Add WeChat powcoder
- (E) balance: 17
- (F) balance: 4

**Answer: ABDF** 

The following is one interleaving that leads to output 0:

- Assignment Project Exam Help
   Thread A executes deposit(4), balance = 14
- Thread B executes with draw (6) palance = 8
- Thread B executes deposit(3), balance = 11
- Thread A executive of the transfer of the t
- Thread B executes withdraw(7), balance = 0

The following is one interleaving that leads to output -3:

- Assignment Project Exam Help
   Thread A executes deposit(4), balance = 14
- Thread A starts to execute with draw (11) and enters the if condition
- Thread B extend to the third to the tendence = 8
- Thread A computes RHS for withdraw(11) = -3
- Thread B executes deposit(3), balance = 11
- Thread A completes withdraw(11), balance = -3
- Thread B executes withdraw(7), balance = -3

The following is one interleaving that leads to output 6:

- Assignment Project Exam Help
   Thread A executes deposit(4), balance = 14
- Thread A exacutes with draw (110 n palance = 3
- Thread B executes withdraw(6), balance = 3
- Thread B executive colleposition, document = 6
- Thread B executes withdraw(7), balance = 6

The following is one interleaving that leads to output 4:

- Assignment Project Exam Help
   Thread B executes withdraw(6), balance = 4
- Thread A executes deposit (4) chalance = 8
- Thread A executes withdraw(11), balance = 8
- Thread B exected test calcinosite (3), doadance = 11
- Thread B executes withdraw(7), balance = 4

## Synchronization



#### Thread Synchronization

How many potential deadlock situations are present?

```
void *thread2(void *vargp) {
void *thread1(void *vargp) {
   V(&add sem);
                      Assignment Project Exam Help
   V(&rem sem);
                                           add();
   remove();
                                           remove();
   P(&add sem);
    P(&rem sem);
                                            int main() {
                                                pthread t tid1, tid2;
    add();
   V(&add sem);
                                                sem init(&rem sem,0,0);
   V(&rem sem);
                                                pthread create(&tid1, NULL, thread1, NULL);
    remove();
                                                pthread create(&tid2, NULL, thread2, NULL);
    add();
                                                pthread join(tid1, NULL);
                                                pthread join(tid2, NULL);
sem t add sem;
                                                return 0;
sem t rem sem;
```

Situation 1:

tid1 executes V(&add\_sem) and V(&rem\_sem). Then, tid2 executes P(&rem\_sem) and P(&add\_sem). In this situation, tid1 can never execute P(&add\_sem) since the value of add\_sem Add AV a Chaulh this oslar deadlock, since after the execution of thread 2, thread 1 can't resume. Thus, there's a deadlock.

Situation 2:

tid1 executes V(&add\_sem) and V(&rem\_sem). Then, tid2 executes P(&nem\_sem). Wextetid1 executes P(&add\_sem). Thread 2 wants to execute P(&add\_sem) but it can't since addlcbehreldsavadoev0odbread 1 wants to execute P(&rem\_sem) but it can't since rem\_sem has value 0. Thus, there's a deadlock.

For lengths 0-6, indicate the number of outcomes of that length that can be produced.

```
sem t add sem;
                            void *thread1(void *vargp) {
 sem t rem sem;
 void add() {
                                                         pthread t tid1, tid2;
      printf("A");
                                                    er. em. t(&add_sem.0.0);
                                                         sem init(&rem sem,0,0);
                                 P(&add sem);
 void remove() {
                                 P(&rem sem);
                                                         pthread create(&tid1, NULL, thread1, NULL);
      printf("R");
                                                        while teate(&tid2, NULL, thread2, NULL);
                                                         pthread join(tid1, NULL);
void *thread2(void *vargp)
                                 V(&add sem);
                                                         pthread join(tid2, NULL);
    P(&rem sem);
                                 V(&rem sem);
    P(&add sem);
                                                         return 0;
                                 remove();
    add();
                                 add();
    remove();
```

Response length 0: None

This is because at least R must get printed due to the call to remove() in thread to be executed by tid1 before any sort of deadlock frond the least that statement gets executed by tid1 before any

Response length 1: 1 (R)

In the deadlock scenario 2, where thread 1 executes P(&add\_sem) and thread 3 executes P(&add\_sem), no print statements are december of the thread 2 executed in pither different that point. The only print statement that gets executed is due to the call to remove() before the calls to P() in thread 1.

Response length 2: None

We noticed that Redue to the call to remove() in thread1() gets printed no/patter what offm the code, we notice that it's not possible for only one other print statement to get executive Chat powcoder

Response length 3: 2 (RAR, ARR)

This happens due to deadlock scenario 1 above, where thread2() executes completely but thread1() can't execute P(&add\_sem) and the statements after that.

#### Add WeChat powcoder

- RAR: Thread 1 executes remove(), followed by thread 2 executing add() and remove().
- ARR: Thread 2 executes add(), followed by any ordering of the 2 calls to remove() by threads 1 and 2.

Response length 4: None

For any length greater than 3, it means that there was no deadlock, since thread 2/could gue to P(), which means it would run to completion was well to P(), which means it length greater than 3 and less than 6 are possible.

Response length 5: None

For any length greater than 3, it means that there was no deadlock, since thread 2/could gure to complete the calls to P(), which means it would run to completed the calls to P(), which means it length greater than 3 and less than 6 are possible.

Response length 6: 4
(RARAAR, RARARA, RAARAR), RASSignment Project Exam Help

Since there are nondeadlocks it means that the initial calls to V() and P() get executed by thread 1. Thus, 'R' and 'A' definitely get the MecClastepthis; observable calls to V() get executed by thread 1 and then, thread 2 can execute its calls to P(). After this, based on the interleavings between the threads, there are 4 possible outputs.

RARAAR: Thread 1 executes remove(), threads 1 and 2 execute the add() statements in any order, and then thread 2 executes remove().

https://powcoder.com RARARA: Thread 1 executes remove(), thread 2 executes add() And d New Check) publicated 1 executes add().

- RAARRA: Thread 2 executes add(), threads 1 and 2 execute the remove() statements in any order, and then thread 1 executes add()! Project Exam Help
- RAARAR: Thread 2 executes add(); thread 1 executes remove() and add(), then thread 2 executes remove().

#### Good luck!



- Logical control Assignment Project Exam Hel
- Private address space

Important system calls

- Fork
- Execve
- Wait
- Waitpid



Draw a Process Graph!!!

Assignment Project Exam Help

(it does not have to be like mine)

https://powcoder.com

Add WeChat powcoder

```
int main() {
                                           What is printed?
   int count = 1;
   int pid1 = fork Assignment Project Exam Help atomic,
   int pid2 = fork();
                                           and all system calls
                      https://powcodesuccend.
   if(pid1 == 0)
       count++;
   else{
                      Add WeChat powcoder
       if(pid2 == 0)
            count--;
       else
         count += 2;
   printf("%d", count);
```

```
int main() {
                                         How many processes?
   int count = 1;
   int pid1 = fork Assignment Project Exam Help
   int pid2 = fork();
                     https://powcoder.com
   if(pid1 == 0)
       count++;
   else{
                     Add WeChat powcoder
       if(pid2 == 0)
            count--;
       else
         count += 2;
   printf("%d", count);
```

```
int main() {
                                             How many processes?
   int count = 1;
   int pid1 = fork Assignment Project Example Helpild
   int pid2 = fork();
                       https://powcoderacennand child: each fork
   if(pid1 == 0)
                                             another child
       count++;
                      Add WeChat powcoder Total: 4 processes
   else{
       if(pid2 == 0)
             count--;
       else
         count += 2;
   printf("%d", count);
```

```
int main() {
                             What does the process diagram look like?
   int count = 1;
   int pid1 = fork Assignment Project Exam Help
   int pid2 = fork();
                     https://powcoder.com
   if(pid1 == 0)
       count++;
   else{
                     Add WeChat powcoder
       if(pid2 == 0)
            count--;
       else
         count += 2;
   printf("%d", count);
```

```
int main() {
                                  What does the process diagram look like?
   int count = 1;
   int pid1 = fork Assignment Project Exam Help
    int pid2 = fork();
                         https://powcoder.com
                                                                Grandchild (from second call to
   if(pid1 == 0)
                                                                fork())
         count++;
                                                                Child1 (from first call to fork())
   else{
                         Add WeChat powcoder
         if(pid2 == 0)
                                                                Child2 (from second call to fork())
               count--;
         else
                                                                        Parent
           count += 2;
   printf("%d", count);
```

```
int main() {
                             What does count look like?
   int count = 1;
   int pid1 = fork Assignment Project Exam Help
   int pid2 = fork();
                             Parent: pid1 != 0 and pid2 != 0
                      https://powpoodenocompid2!=0
   if(pid1 == 0)
                             Child2: pid1 != 0 and pid2 == 0
       count++;
                      Add Wardshild: pid weed pid2 == 0
   else{
       if(pid2 == 0)
             count--;
       else
         count += 2;
   printf("%d", count);
```

```
int main() {
                             What does count look like?
   int count = 1;
   int pid1 = fork Assignment Project Exam Help
   int pid2 = fork();
                             Parent: pid1 != 0 and pid2 != 0
                      https://powcoder.com
   if(pid1 == 0)
                             Child1: pid1 == 0 and pid2 != 0
       count++;
                      Add WeChaphbacoder
   else{
       if(pid2 == 0)
                             Child2: pid1 != 0 and pid2 == 0
            count--;
                                     count = 0
       else
                             Grandchild: pid1 == 0 and pid2 == 0
         count += 2;
                                      count = 2
   printf("%d", count);
```

```
int main() {
                                     Given the process diagram, what are the
    int count = 1;
    different permutations that can be printed int pid2 = fork():

different permutations that can be printed int pid2 = fork():
    int pid2 = fork();
                            https://powcoder.com
    if(pid1 == 0)
         count++;
                                                                                count = 2
                            Add WeChat powcoder
    else{
         if(pid2 == 0)
                                                                                count = 2
                count--;
         else
                                                                                count = 0
           count += 2;
    printf("%d", count);
                                                                                count = 3
```

```
Given the process diagram, what are the
                               different permutations that can be printed
int main() {
   int count = 1;
                               out?
   int pid1 = forkAssignment Project Exam Help
   int pid2 = fork();
                       https://powcoder.dom
                                                                    count = 2
   if(pid1 == 0)
        count++;
                                                                    count = 2
                       Add WeChat powcoder
   else{
        if(pid2 == 0)
                                                                    count = 0
             count--;
        else
                                                                    count = 3
          count += 2;
   printf("%d", count);
                                   Math!
                                            4! / 2 = 12 different possible outcomes
```

#### Remember:

• Processes can occur in Assignment Project Examily of Processes can occur in

Watch out for a wait or

https://powcoder.com waitpid!

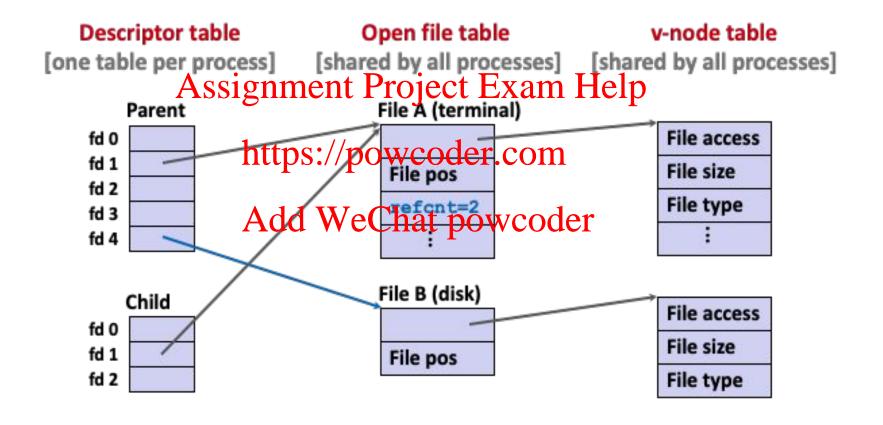
What if I included a

Add WeChat powcoder | wait (NULL) before | printed out count?

Good luck!



### **How the Unix Kernel Represents Open Files**



#### Main ideas:

```
foo.txt: abcdefgh...xyz
int main() {
    int fd1, fd2, fd3;
    char c;
    pid_t pid;
    fd1 = open("foo.txt", O_RDONLY);
    fd2 = open("foo.txt", D_RDONLY);
    fd3 = open("foo.txt", DEDONLY);
    read(fd1, &c, sizeof(c));
    read(fd2, &c, sizeof(c));
    read(fd3, &c, sizeof(c));
    read(fd3, &c, sizeof(c));
    read(fd2, &c, sizeof(c));
    read(fd2,
```

```
foo.txt: abcdefqh...xyz
                                                How does read offset?
int main() {
                                                     Incremented by number of bytes
    int fd1, fd2, fd3;
                   Assignment Project Exam Help How does dup2 work?
    char c;
    pid t pid;
    fd1 = open("foo.txt", O RDONLY);
                                                  Any read/write from fd3 now
    fd2 = open("foo.txt", O RDONLY);
                                                der. Gappan from fd2
    fd3 = open("foo.txt", propsy)/;/
    read(fd1, &c, sizeof(c));
                                                     All file offsets are shared
    read(fd2, &c, sizeof(c));
    dup2(fd2, fd3); read(fd3, &c, sizeof(e)), Add WeChatpowcoder
     read(fd2, &c, sizeof(c));
```

```
Main ideas:
read(fd1, &c, sizeof(c)); // a
read(fd2, &c, sizeof(c)); // a
                                        How are fd shared between
dup2(fd2, fd3);
read (fd3, &c, sizes signment Project Example)
                                        How does dup2 work from parent
                                        to child?
pid = fork();
if (pid==0) {
                                               elfile offsets shared
     read(fd1, &c, sizeof(c));
                                        between processes?
     printf("c = %c\n", c):
     dup2(fd1, fd2):Add
     read(fd3, &c, sizeof(c));
     printf("c = %c\n", c);
read(fd2, &c, sizeof(c));
printf("c = %c\n", c);
read(fd1, &c, sizeof(c));
printf("c = %c\n'', c);
```

```
read(fd1, &c, sizeof(c)); // a
read(fd2, \&c, sizeof(c)); // a
dup2(fd2, fd3);
read (fd3, &c, sixes fighment Project Exam Help
                                      What would this program print?
pid = fork();
                    https://powcoder.com
Just ignore the possible outcomes due
if (pid==0) {
    read(fd1, &c, sizeof(c));
                                      to interleaving ... try two simple cases:
    printf("c = %c\n", c); dup2(fd1, fd2); \overrightarrow{Add}
    read(fd3, &c, sizeof(c));
                                       1. First child executes to the end
    printf("c = %c\n", c);
                                           First parent executes to the end.
read(fd2, &c, sizeof(c));
printf("c = %c\n", c);
read(fd1, &c, sizeof(c));
printf("c = %c\n", c);
```

```
read(fd1, \&c, sizeof(c)); // a
                                   Possible output 1:
read(fd2, &c, sizeof(c)); // a
                                       c = b // in child
dup2(fd2, fd3);
read (fd3, &c, sixes fignment Project Examchelp
                                       c = c // in child
pid = fork();
if (pid==0) {
    read(fd1, &c, sizeof(c));
                                   c = e // in parent
    printf ("c = %c\n", c) we Chat proved in parent
    read(fd3, &c, sizeof(c));
    printf("c = %c\n", c);
read(fd2, &c, sizeof(c));
printf("c = %c\n", c);
read(fd1, &c, sizeof(c));
printf("c = %c\n", c);
```

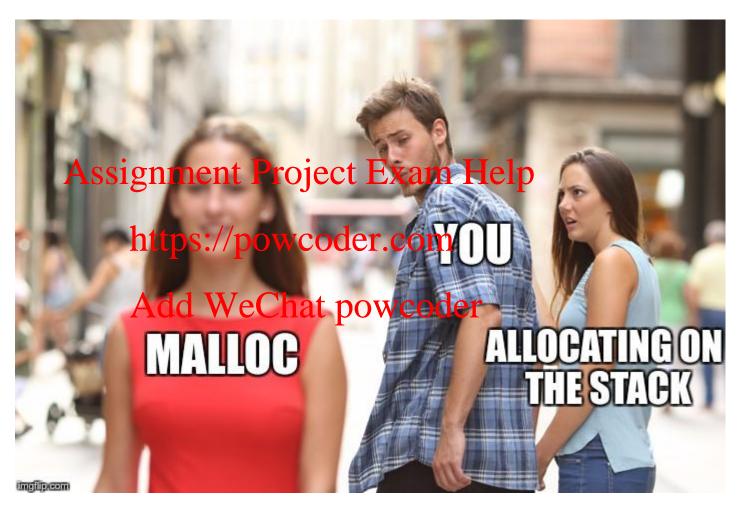
```
read(fd1, \&c, sizeof(c)); // a
                                   Possible output 2:
read(fd2, &c, sizeof(c)); // a
                                       c = d // in parent
dup2(fd2, fd3);
read (fd3, &c, sixes figure ent Project Examplicate)
                                       c = c // in child from fd1
pid = fork();
                   https://powcoder &pimchild from fd3
if (pid==0) {
    read(fd1, &c, sizeof(c));
                                       c = d // in child
    dup2 (fd1, fd2) Add WeChat povecoderild
    read(fd3, &c, sizeof(c));
    printf("c = %c\n", c);
read(fd2, &c, sizeof(c));
printf("c = %c\n", c);
read(fd1, &c, sizeof(c));
printf("c = %c\n", c);
```

```
pid = fork();
if (pid==0) {
   read (fd1, Assignment, Project Exam Help
   printf("c = %c\n", c); What are the possible outputs now?
   dup2(fd1, fd2);https://powcoder.com
   printf("c = %c\n'', c);
if (pid!=0) waitpid Add W.E. hat powcoder
read(fd2, &c, sizeof(c));
printf("c = %c\n", c);
read(fd1, &c, sizeof(c));
printf("c = %c\n", c);
return 0;
```

```
Possible output:
pid = fork();
                                      c = b // in child
if (pid==0)
    read (fd1, Assignment Project ExamcHelp
    printf("c = %c\n", c);
                                      c = c // in child
    dup2(fd1, fd2);https:
    read(fd3, &c, size of %c)
    printf("c = %c\n", c);
                                      c = e // in parent
if (pid!=0) waitpid Add W. e Chat poved in learent
read(fd1, &c, sizeof(c));
printf("c = %c\n", c);
read(fd2, &c, sizeof(c));
printf("c = %c\n", c);
return 0;
```

```
Child creates a copy of the parent
read(fd1, \&c, sizeof(c)); // a
read(fd2, &c, sizeof(c)); // a
                                     fd table
dup2(fd2, fd3);
read (fd3, &c, sixes figh ment Project Example the child
                                         dup2/open/close in child do
pid = fork();
                  https://powcoder.comaffect the parent
if (pid==0) {
    read(fd1, &c, sizeof(c));
                                  File descriptors across process
    dup2(fd1, fd2);
    read (fd3, &c, Add (WeChat phototheleame file offset.
if (pid!=0) waitpid(-1, NULL, 0);
read(fd2, &c, sizeof(c));
read(fd1, &c, sizeof(c));
```

#### Malloc



#### Malloc

- Fit algorithms first/next/best/good
- Fragmentation

  Assignment Project Exam Help
  Internal inside blocks

  - External between blocksder.com
- Organization
  - Implicit Add WeChat powcoder
  - **Explicit**
  - Segregated







Ma	lloc -	First	fit
IVIC		1 11 5 6	

- 16 byte align
- coalesced Assignfooterless
- 32 min size

	#1	#2	#3	#4	#5	#6
a = malloc(32)						
b = malloc(16)						
menti Proj	ect E	xam	Help			
d = malloc(40) tps://powo free(c)	oder	com				
dd Weena	ıt pov	vcode	er			
e = malloc(16)	•					
free(d)						
f = malloc(48)						
free(b)						
	b = malloc(16)  nematice(ro)  d = malloc(40)  ps://powcfree(c)  dd Weree(h)  e = malloc(16)  free(d)  f = malloc(48)	a = malloc(32) b = malloc(16)  mendideroject E  d = malloc(40) ps://powcodero free(c)  d Weelaat pov e = malloc(16)  free(d)  f = malloc(48)	a = malloc(32)  b = malloc(16)  mentilero ect Exam  d = malloc(40) ps://powcoder.com  free(c)  dd Weelhat powcode  e = malloc(16)  free(d)  f = malloc(48)	a = malloc(32)  b = malloc(16)  mention ect Exam Help  d = malloc(40) ps://powcoder.com  free(c)  dd Weehat powcoder  e = malloc(16)  free(d)  f = malloc(48)	a = malloc(32) b = malloc(16)  mentileroj ect Exam Help d = malloc(40) ps://powcoder.com free(c)  dd Weehat powcoder e = malloc(16)  free(d) f = malloc(48)	a = malloc(32) b = malloc(16)  mentileroject Exam Help  d = malloc(40) ps://powcoder.com free(c)  dd Weelhat powcoder  e = malloc(16)  free(d)  f = malloc(48)

<b>M</b> /2	alld	C	_	Fi	rs	t f	it
IVIC	コル		_		I O	LI	IL

- 16 byte align
- coalesced Assignmentfooterless
- 32 min size

t		#1	#2	#3	#4	#5	#6
	a = malloc(32)	48a					
	b = malloc(16)						
gr	menti Proj	ect E	xam	Help			
ht	d = malloc(40) tps://powo free(c)	oder.	com				
A	dd Weena	ıt pov	vcode	er			
	e = malloc(16)	4					
	free(d)						
	f = malloc(48)						
	free(b)						

Ma	lloc -	First	fit
IVIC		1 11 5 6	

- 16 byte align
- coalesced Assignfooterless
- 32 min size

	#1	#2	#3	#4	#5	#6
a = malloc(32)	48a					
b = malloc(16)	48a	32a				
nmenti Proj	ect E	xam	Help			
d = malloc(40) ttps://powcfree(c)	oder	com				
dd Weeha	ıt pov	vcode	er			
e = malloc(16)	4					
free(d)						
f = malloc(48)						
free(b)						

Ma	lloc -	First	fit
IVIC		1 11 5 6	

- 16 byte align
- coalesced Assignfooterless
- 32 min size

		#1	#2	#3	#4	#5	#6
	a = malloc(32)	48a					
	b = malloc(16)	48a	32a				
n	menti Proj	eeta E	Xæn	Help			
tit	d = malloc(40)  ps://powc free(c)	oder	com				
	ld Weena	ıt pov	vcode	er			
	e = malloc(16)	4					
	free(d)						
	f = malloc(48)						
	free(b)						

Mal		First	fit
IVIal	10C -	ΓIISt	ΗL

- 16 byte align
- coalesced Assignfooterless
- 32 min size

	#1	#2	#3	#4	#5	#6
a = malloc(32)	48a					
b = malloc(16)	48a	32a				
nment Proj	eeta E	X	Help			
d = malloc(40)	48a	32a	32a	48a		
free(c)	<del>oder</del> .	com				
dd Weena	ıt pov	vcode	er			
e = malloc(16)	1					
free(d)						
f = malloc(48)						
free(b)						

- 16 byte align
- coalesced Assignfooterless
- 32 min size

		#1	#2	#3	#4	#5	#6
	a = malloc(32)	48a					
	b = malloc(16)	48a	32a				
n	menti Proj	eeta E	X	Help			
4	d = malloc(40)	48a	32a	32a	48a		
u	ps://powc free(c)	48a	COM 32a	32f [0]	48a		
1	dd Weena	ıt pov	vcode	er			
	e = malloc(16)	4					
Ī	free(d)						
	f = malloc(48)						
	free(b)						

- 16 byte align
- coalesced Assignfooterless
- 32 min size

	#1	#2	#3	#4	#5	#6
a = malloc(32)	48a					
b = malloc(16)	48a	32a				
nment Proj	eeta E	X	Help			
d = malloc(40)	48a	32a	32a	48a		
free(c)	48a	COM 32a	32f [0]	48a		
dd Weena	14850V	vædde	<b>2</b> 2f [1]	48a		
e = malloc(16)	4					
free(d)						
f = malloc(48)						
free(b)						

- 16 byte align
- coalesced Assignfooterless
- 32 min size

		#1	#2	#3	#4	#5	#6
	a = malloc(32)	48a					
	b = malloc(16)	48a	32a				
r	menti Proj	eeta E	Xæn	Help			
4	d = malloc(40)	48a	32a	32a	48a		
L	free(c)	48a	COM 32a	32f [0]	48a		
V	dd Weena	1 <mark>4856</mark> 0	vædde	<b>2</b> 2f [1]	48a		
	e = malloc(16)	48a	32a	32f [0]	48a		
	free(d)						
	f = malloc(48)						
	free(b)						

- 16 byte align
- coalesced Assignfooterless
- 32 min size

		#1	#2	#3	#4	#5	#6
	a = malloc(32)	48a					
	b = malloc(16)	48a	32a				
<b>5</b> 1	menti Proj	eeta E	Xæn	Help			
٠,	d = malloc(40)	48a	32a	32a	48a		
Il	free(c)	48a	COM 32a	32f [0]	48a		
4	dd Weena	1 <mark>485</mark> 60V	vædde	<b>2</b> 2f [1]	48a		
	e = malloc(16)	48a	32a	32f [0]	48a		
	free(d)	48a	32a	80f [0]			
	f = malloc(48)						
	free(b)						

- 16 byte align
- coalesced Assignfooterless
- 32 min size

		#1	#2	#3	#4	#5	#6
	a = malloc(32)	48a					
	b = malloc(16)	48a	32a				
1	menti Proj	eeta E	Xæn	Help			
4	d = malloc(40)	48a	32a	32a	48a		
Ų	free(c)	Oder 48a	COM 32a	32f [0]	48a		
\	dd Weena	1 <mark>4851</mark> 87	vædde	2f [1]	48a		
	e = malloc(16)	48a	32a	32f [0]	48a		
	free(d)	48a	32a	80f [0]			
	f = malloc(48)	48a	32a	80a			
	free(b)						

- 16 byte align
- coalesced Assignfooterless
- 32 min size

	#1	#2	#3	#4	#5	#6
a = malloc(32)	48a					
b = malloc(16)	48a	32a				
ımentı Proj	eeta E	Xæn	Help			
d = malloc(40)	48a	32a	32a	48a		
free(c)	48a	32a	32f [0]	48a		
dd Weena	լ <del>(</del> 485/6)Ն	vædde	2f [1]	48a		
e = malloc(16)	48a	32a	32f [0]	48a		
free(d)	48a	32a	80f [0]			
f = malloc(48)	48a	32a	80a			
free(b)	48a	32f [0]	80a			
	b = malloc(16)  nematice(ro)  d = malloc(40)  tps://powc free(c)  dd Weree(b)  e = malloc(16)  free(d)  f = malloc(48)	a = malloc(32) 48a  b = malloc(16) 48a  menalloc(40) 48a  d = malloc(40) 48a  tps://powoder 48a  dd Weree(a) 148500  e = malloc(16) 48a  free(d) 48a  f = malloc(48) 48a	a = malloc(32) 48a  b = malloc(16) 48a 32a  mendicero esta Exam  d = malloc(40) 48a 32a  tps://powcoder com free(c) 48a 32a  dd Weela table weela 2a  e = malloc(16) 48a 32a  free(d) 48a 32a  free(d) 48a 32a  free(d) 48a 32a	a = malloc(32) 48a  b = malloc(16) 48a 32a  mendictro esta Exam Heap  d = malloc(40) 48a 32a 32a  tps://powcoder com free(c) 48a 32a 32f [0]  dd Weelha table weelha 32a 32f [0]  e = malloc(16) 48a 32a 32f [0]  free(d) 48a 32a 80f [0]  f = malloc(48) 48a 32a 80a	a = malloc(32) 48a  b = malloc(16) 48a 32a  maematicProjecta Exam Heap  d = malloc(40) 48a 32a 32a 48a  tps://powoder com 32a 32f [0] 48a  dd Weelba 14856 weeld 22f [1] 48a  e = malloc(16) 48a 32a 32f [0] 48a  free(d) 48a 32a 80f [0]  f = malloc(48) 48a 32a 80a	a = malloc(32) 48a  b = malloc(16) 48a 32a  maenal Cro esta E x 22an Heap  d = malloc(40) 48a 32a 32a 48a  ps://powcoder com 32a 32f [0] 48a  dd Weelaa 148 10 w 32a 22f [1] 48a  e = malloc(16) 48a 32a 32f [0] 48a  free(d) 48a 32a 80f [0]  f = malloc(48) 48a 32a 80a

- 16 byte align
- coalesced Assign
- footerless\*
- 32 min size
- fragmentation?
  - internal?

	#1	#2	#3	#4	#5	#6
a = malloc(32)	48a					
b = malloc(16)	48a	32a				
ımentı Aroj	eeta E	X	Help			
d = malloc(40)	48a	32a	32a	48a		
free(c)	48a	COM 32a	32f [0]	48a		
dd Weena	1 <mark>48516</mark> 17	vædde	2f [1]	48a		
e = malloc(16)	48a	32a	32f [0]	48a		
free(d)	48a	32a	80f [0]			
f = malloc(48)	48a	32a	80a			
free(b)	48a	32f [0]	80a			

- 16 byte align
- coalesced Assignfooterless
- 32 min size
- fragmentation?
  - internal
  - **(48-16) +** (80-48) = 64
  - external?

	#1	#2	#3	#4	#5	#6
a = malloc(32)	48a					
b = malloc(16)	48a	32a				
ment Proj	eeta E	X	Help			
d = malloc(40)	48a	32a	32a	48a		
free(c)	48a	COM 32a	32f [0]	48a		
dd Weeha	1 <mark>48516</mark> 1V	vædde	<b>2</b> 2f [1]	48a		
e = malloc(16)	48a	32a	32f [0]	48a		
free(d)	48a	32a	80f [0]			
f = malloc(48)	48a	32a	80a			
free(b)	48a	32f [0]	80a			

- 16 byte align
- coalesced
- footerless
- 32 min size
- fragmentation?
  - internal
  - **(48-16) +** (80-48) = 64
  - external
  - 32

	#1	#2	#3	#4	#5	#6
a = malloc(32)	48a					
b = malloc(16)	48a	32a				
ımentı Aroj	eeta E	Xæan	Help			
d = malloc(40)	48a	32a	32a	48a		
free(c)	48a	COM 32a	32f [0]	48a		
dd Weena	1 <mark>485</mark> 69V	vædde	<b>2</b> 2f [1]	48a		
e = malloc(16)	48a	32a	32f [0]	48a		
free(d)	48a	32a	80f [0]			
f = malloc(48)	48a	32a	80a			
free(b)	48a	32f [0]	80a			

Ν/	all	_ R	est	fit
IVI	all	<b>-</b> D	けるし	IIL

- 16 byte align
- coalesced Assignfooterless
- 32 min size

		#1	#2	#3	#4	#5	#6
a = ma	alloc(32)						
b = ma	alloc(16)						
gnmena	tı <b>P</b> (roj	ect E	xam	Help			
d = ma nttps://	plloc(40) powc free(c)	oder	com				
Add W	eeh2	ıt pov	vcode	er			
e = ma	alloc(16)						
	free(d)						
f = ma	alloc(48)						
	free(b)						

Ν/	all	_ R	est	fit
IVI	all	<b>-</b> D	けるし	IIL

- 16 byte align
- coalesced Assignfooterless
- 32 min size

		#1	#2	#3	#4	#5	#6
	a = malloc(32)	48a					
	b = malloc(16)	48a	32a				
ŋ	menti Proj	eeta E	Xæan	Help			
4	d = malloc(40)	48a	32a	32a	48a		
Ų	free(c)	48a	COM 32a	32f [0]	48a		
7	dd Weena	1 <mark>4856</mark> 0	vædde	<b>2</b> 2f [1]	48a		
	e = malloc(16)	4					
	free(d)						
	f = malloc(48)						
	free(b)						

- 16 byte align
- coalesced Assignfooterless
- 32 min size

		#1	#2	#3	#4	#5	#6
	a = malloc(32)	48a					
	b = malloc(16)	48a	32a				
<b>51</b>	menti Proj	eeta E	Xæn	Help			
<b>.</b> +	d = malloc(40)	48a	32a	32a	48a		
ll	free(c)	OGCT 48a	COM 32a	32f [0]	48a		
1	dd Weena	1 <mark>48516</mark> 17	vædde	<b>2</b> 2f [1]	48a		
	e = malloc(16)	48f [0]	32a	32a	48a		
	free(d)						
	f = malloc(48)						
	free(b)						

NΛ	ااد	_ R	est	fit
IVI	all	<b>-</b> D	けるし	IIL

- 16 byte align
- coalesced Assignfooterless
- 32 min size

		#1	#2	#3	#4	#5	#6
	a = malloc(32)	48a					
	b = malloc(16)	48a	32a				
r	menti Proj	eeta E	Xæn	Help			
4	d = malloc(40)	48a	32a	32a	48a		
L	tps://powc	48a	COM 32a	32f [0]	48a		
V	dd Weena	1 <mark>485</mark> 60v	vædde	<b>2</b> 2f [1]	48a		
	e = malloc(16)	48f [0]	32a	32a	48a		
	free(d)	48f [1]	32a	32a	48f [0]		
	f = malloc(48)						
	free(b)						

- 16 byte align
- coalesced Assignfooterless
- 32 min size

	#1	#2	#3	#4	#5	#6
a = malloc(32)	48a					
b = malloc(16)	48a	32a				
nment Proj	eeta E	Xæan	Help			
d = malloc(40)	48a	32a	32a	48a		
free(c)	48a	COM 32a	32f [0]	48a		
dd Weena	1 <mark>48516</mark> 1V	vædde	<b>2</b> 2f [1]	48a		
e = malloc(16)	48f [0]	32a	32a	48a		
free(d)	48f [1]	32a	32a	48f [0]		
f = malloc(48)	48f [0]	32a	32a	64a		
free(b)						

- 16 byte align
- coalesced Assignfooterless
- 32 min size

	#1	#2	#3	#4	#5	#6
a = malloc(32)	48a					
b = malloc(16)	48a	32a				
nmenti Proj	eeta E	Xæn	Help			
d = malloc(40)	48a	32a	32a	48a		
free(c)	48a	COM 32a	32f [0]	48a		
dd Weeha	լ <del>(</del> 48 <b>5</b> (8) Ն	vædde	2f [1]	48a		
e = malloc(16)	48f [0]	32a	32a	48a		
free(d)	48f [1]	32a	32a	48f [0]		
f = malloc(48)	48f [0]	32a	32a	64a		
free(b)	80f	[0]	32a	64a		

- 16 byte align
- coalesced Assign
- footerless\*
- 32 min size
- fragmentation?
  - internal?

	#1	#2	#3	#4	#5	#6
a = malloc(32)	48a					
b = malloc(16)	48a	32a				
ımentı Aroj	eeta E	X	Help			
d = malloc(40)	48a	32a	32a	48a		
free(c)	48a	COM 32a	32f [0]	48a		
dd Weena	1 <mark>485</mark> 60V	væde	<b>2</b> 2f [1]	48a		
e = malloc(16)	48f [0]	32a	32a	48a		
free(d)	48f [1]	32a	32a	48f [0]		
f = malloc(48)	48f [0]	32a	32a	64a		
free(b)	80f	[0]	32a	64a		

- 16 byte align
- coalesced Assignfooterless
- 32 min size
- fragmentation?
  - internal
  - **(32-16) +** (64-48) = 32
  - external?

	#1	#2	#3	#4	#5	#6
a = malloc(32)	48a					
b = malloc(16)	48a	32a				
nment Proj	eeta E	Xæan	Help			
d = malloc(40)	48a	32a	32a	48a		
free(c)	48a	COM 32a	32f [0]	48a		
dd Weena	1 <mark>485</mark> 69V	vædde	<b>2</b> 2f [1]	48a		
e = malloc(16)	48f [0]	32a	32a	48a		
free(d)	48f [1]	32a	32a	48f [0]		
f = malloc(48)	48f [0]	32a	32a	64a		
free(b)	80f	[0]	32a	64a		

- 16 byte align
- coalesced Assignfooterless
- 32 min size
- fragmentation?
  - internal
  - **(32-16) +** (64-48) = 32
  - external
  - 80

	#1	#2	#3	#4	#5	#6
a = malloc(32)	48a					
b = malloc(16)	48a	32a				
nmenti Proj	eeta E	Xæan	Help			
d = malloc(40)	48a	32a	32a	48a		
free(c)	48a	COM 32a	32f [0]	48a		
dd Weena	1 <mark>4856</mark> 0	vædde	<b>2</b> 2f [1]	48a		
e = malloc(16)	48f [0]	32a	32a	48a		
free(d)	48f [1]	32a	32a	48f [0]		
f = malloc(48)	48f [0]	32a	32a	64a		
free(b)	80f	[0]	32a	64a		

# Signals

# who would win? Assignment Project Exam Help

several hundred lines of one asynchronous boi tshlab codeps://powcoder.com



## Signals

if (child == 0) {

exit(0);

/\* insert code here \*/

sleep(1); waitpid(child, NULL, 0);

printf("Received %d USR{1,2} signals\n", counter);

Child calls kill(parent, SIGUSR{1,2}) between 2-4 times.
What sequence of kills may print 1?
Can you guarantee printing 2?
What is the range devalue printer oject Exam Help
int counter = 0;
void handler (int sig) {
 atomically {counter++;} https://powcoder.com
}
int main(int argc, char\*\* argv) {
 signal(SIGUSR1, handler); Add WeChat powcoder
 int parent = getpid(); int child = fork();

# Signals (Contd.)

Sending the same signal to the parent in all the calls to kill() may print 1 since there would be no queuing of signals.

- https://powcoder.com
  We can guarantee printing 2 if we send precisely one SIGUSR1 and Antel SMG-USBR2 powcoder
- We can print 1-4 depending on the manner in which signals are sent and received.