SER 334 A Session

Exam 1 Review Session

Thursday, January 25th 2024

7:00 pm - 9:00 pm MST

Agenda

{
Pointer Tracing

Sample Problems

Practice Exam

SI Session Expectations

Thanks for coming to the **SER 334** SI session. We have a packed agenda and we are going to try to get through as many of our planned example problems as possible. This session will be recorded and shared with others.

- If after this you want to see additional examples, please visit the drop-in tutoring center.
- We will post the link in the chat now and at the end of the session.
  - tutoring.asu.edu
- Please keep in mind we are recording this session and it will be made available for you to review 24-48 hours after this session concludes.
- Finally, please be respectful to each other during the session.

Interact with us:
Zoom Features

Zoom Chat

- Use the chat feature to interact with the presenter and respond to presenter's questions.
- Annotations are encouraged
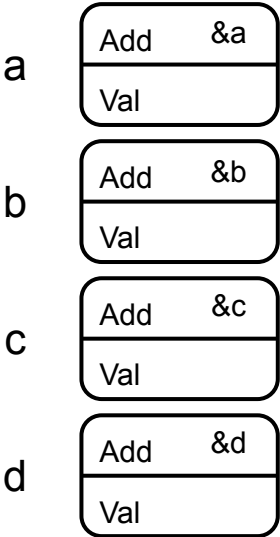
SER 334
Exam Tips

Question 1

- Read each question carefully
- Use the scratch spaces!!
- These Exams are testing your knowledge ***in addition*** to your ability to prioritize tasks and manage your time
- Don't spend too much time on one question!
  - If you get stuck, select the answer you think and mark the question
  - Come back *after the programming questions* to re-attempt
- Pay attention to the clock!
  - Make sure to leave yourself enough time for programming!

Unit 1, Question 8

SER 334
Sample Review

```
1   int a, *b, **c, d;
2   a = 5;
3   d = a + 1;
4   b = &a;
5   c = &b
6   // Point 1
7   *b = 8;
8   b = &d;
9   **c = 3;
10  // Point 2
11   b+=7;
12   // Point 3
```
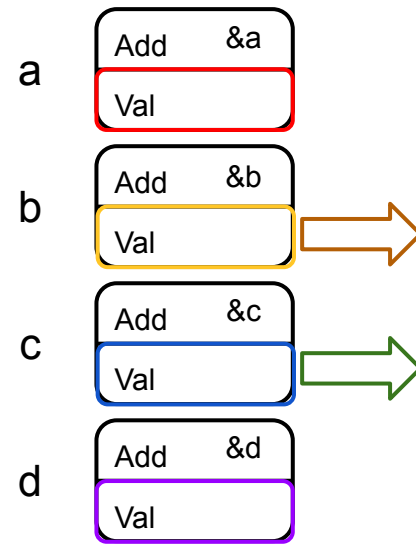
a | Add &a | Val
b | Add &b | Val
c | Add &c | Val
d | Add &d | Val

Type	int	int (deref)	int pointer	int pointer (deref)	int double pointer	int
Var Name	a	*b	b	*c	c	d
Point 1						
Point 2						
Point 3						

SER 334
Sample Review

```
1    int a, *b, **c, d;
2    a = 5;
3    d = a + 1;
4    b = &a;
5    c = &b
6    // Point 1
7    *b = 8;
8    b = &d;
9    **c = 3;
10   // Point 2
11    b+=7;
12     // Point 3
```

a

Add	&a
Val	

b

Add	&b
Val	

c

Add	&c
Val	

d

Add	&d
Val	

Type	int	int (deref)	int pointer	int pointer (deref)	int double pointer	int
Var Name	a	*b	b	*c	c	d
Point 1						
Point 2						
Point 3						

SER 334
Sample Review

```
1    int a, *b, **c, d;
2    a = 5;
3    d = a + 1;
4    b = &a;
5    c = &b
6    // Point 1
7    *b = 8;
8    b = &d;
9    **c = 3;
10   // Point 2
11    b+=7;
12     // Point 3
```

a
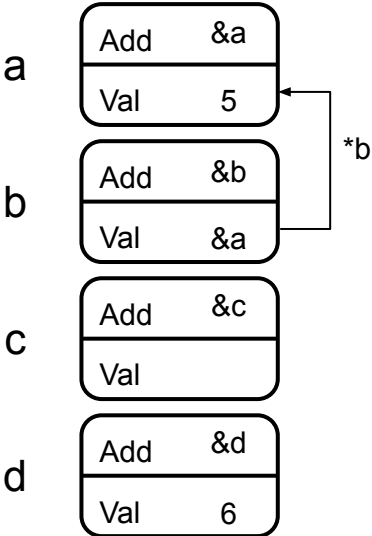| Add | &a |
| Val | 5 |

b
| Add | &b |
| Val | |

c
| Add | &c |
| Val | |

d
| Add | &d |
| Val | |

Type	int	int (deref)	int pointer	int pointer (deref)	int double pointer	int
Var Name	a	*b	b	*c	c	d
Point 1						
Point 2						
Point 3						

SER 334
Sample Review
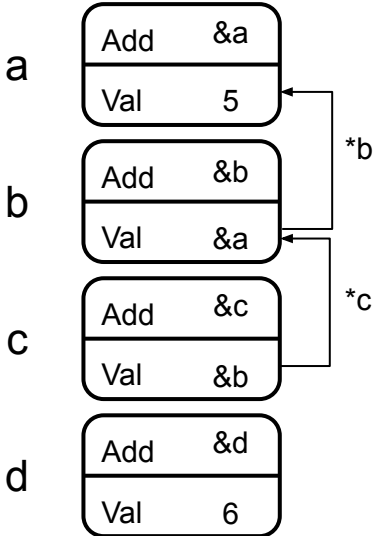
```
1    int a, *b, **c, d;
2    a = 5;
3    d = a + 1;
4    b = &a;
5    c = &b
6    // Point 1
7    *b = 8;
8    b = &d;
9    **c = 3;
10   // Point 2
11    b+=7;
12     // Point 3
```

a
Add	&a
Val	5

b
Add	&b
Val	

c
Add	&c
Val	

d
Add	&d
Val	6

Type	int	int (deref)	int pointer	int pointer (deref)	int double pointer	int
Var Name	a	*b	b	*c	c	d
Point 1	5					
Point 2						
Point 3						

SER 334
Sample Review
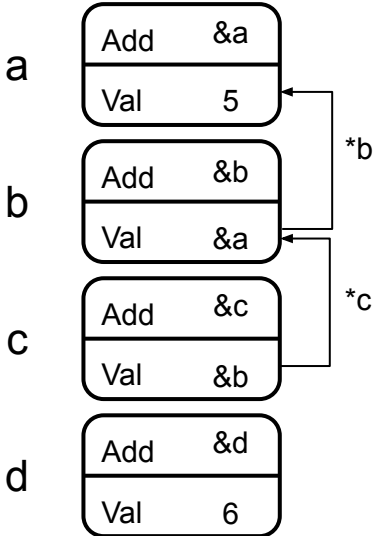
```
1    int a, *b, **c, d;
2    a = 5;
3    d = a + 1;
4    b = &a;
5    c = &b
6    // Point 1
7    *b = 8;
8    b = &d;
9    **c = 3;
10   // Point 2
11    b+=7;
12     // Point 3
```

a
| Add | &a |
| Val | 5 |

*b

b
| Add | &b |
| Val | &a |

c
| Add | &c |
| Val | |

d
| Add | &d |
| Val | 6 |

Type	int	int (deref)	int pointer	int pointer (deref)	int double pointer	int
Var Name	a	*b	b	*c	c	d
Point 1	5					6
Point 2						
Point 3						

SER 334
Sample Review
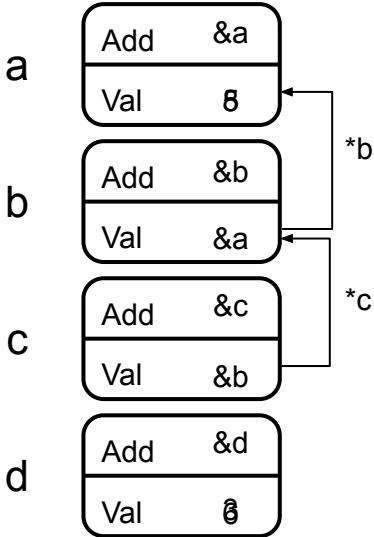
```
1    int a, *b, **c, d;
2    a = 5;
3    d = a + 1;
4    b = &a;
5    c = &b
6    // Point 1
7    *b = 8;
8    b = &d;
9    **c = 3;
10   // Point 2
11    b+=7;
12    // Point 3
```

a
Add	&a
Val	5

*b

b
Add	&b
Val	&a

*c

c
Add	&c
Val	&b

d
Add	&d
Val	6

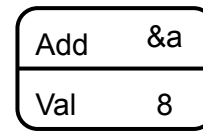| Type	int	int (deref)	int pointer	int pointer (deref)	int double pointer	int
Var Name	a	*b	b	*c	c	d
Point 1	5		Add. of a			6
Point 2						
Point 3						

SER 334
Sample Review
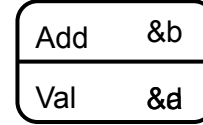
```
1    int a, *b, **c, d;
2    a = 5;
3    d = a + 1;
4    b = &a;
5    c = &b
6    // Point 1
7    *b = 8;
8    b = &d;
9    **c = 3;
10   // Point 2
11    b+=7;
12    // Point 3
```

a

Add	&a
Val	5

*b

b

Add	&b
Val	&a

*c

c

Add	&c
Val	&b

d

Add	&d
Val	6

Type	int	int (deref)	int pointer	int pointer (deref)	int double pointer	int
Var Name	a	*b	b	*c	c	d
Point 1	5		Add. of a			6
Point 2						
Point 3						

SER 334
Sample Review

```
1    int a, *b, **c, d;
2    a = 5;
3    d = a + 1;
4    b = &a;
5    c = &b
6    // Point 1
7    *b = 8;
8    b = &d;
9    **c = 3;
10   // Point 2
11    b+=7;
12    // Point 3
```
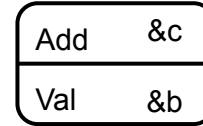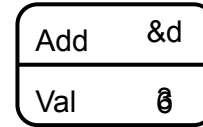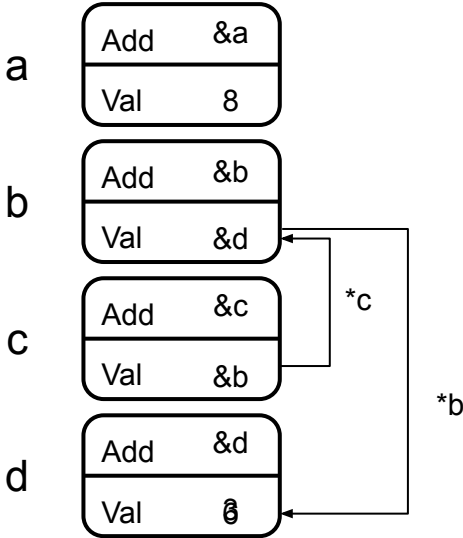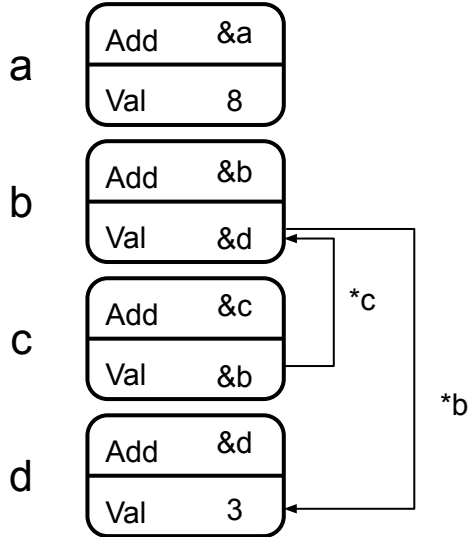
a
Add	&a
Val	8 5

*b

b
Add	&b
Val	&a

*c

c
Add	&c
Val	&b

d
Add	&d
Val	6 3

Type	int	int (deref)	int pointer	int pointer (deref)	int double pointer	int
Var Name	a	*b	b	*c	c	d
Point 1	5	5	Add. of a	Add. of a	Add. of b	6
Point 2						
Point 3						

SER 334
Sample Review
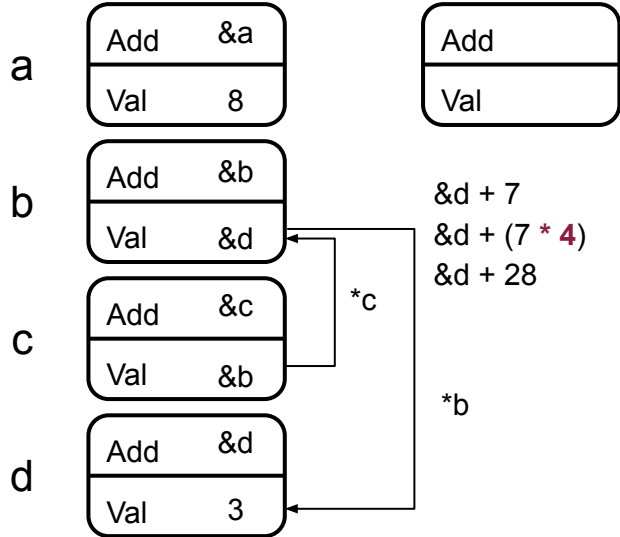
```
1    int a, *b, **c, d;
2    a = 5;
3    d = a + 1;
4    b = &a;
5    c = &b
6    // Point 1
7    *b = 8;
8    b = &d;
9    **c = 3;
10   // Point 2
11    b+=7;
12    // Point 3
```

a

Add	&a
Val	8

*b

b

Add	&b
Val	&d~~a~~

*c

c

Add	&c
Val	&b

*b

d

Add	&d
Val	~~6~~ ~~8~~ 3

Type	int	int (deref)	int pointer	int pointer (deref)	int double pointer	int
Var Name	a	*b	b	*c	c	d
Point 1	5	5	Add. of a	Add. of a	Add. of b	6
Point 2	8					
Point 3						

SER 334
Sample Review

```
1    int a, *b, **c, d;
2    a = 5;
3    d = a + 1;
4    b = &a;
5    c = &b
6    // Point 1
7    *b = 8;
8    b = &d;
9    **c = 3;
10   // Point 2
11    b+=7;
12    // Point 3
```

a
| Add | &a |
| Val | 8 |

b
| Add | &b |
| Val | &d |

c
| Add | &c |
| Val | &b |

*c

*b

d
| Add | &d |
| Val | 6 3 |

Type	int	int (deref)	int pointer	int pointer (deref)	int double pointer	int
Var Name	a	*b	b	*c	c	d
Point 1	5	5	Add. of a	Add. of a	Add. of b	6
Point 2	8		Add. of d			
Point 3						

SER 334
Sample Review
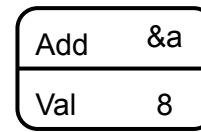
```
1    int a, *b, **c, d;
2    a = 5;
3    d = a + 1;
4    b = &a;
5    c = &b
6    // Point 1
7    *b = 8;
8    b = &d;
9    **c = 3;
10   // Point 2
11     b+=7;
12      // Point 3
```
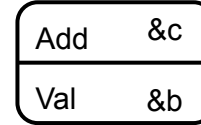
a
Add	&a
Val	8

b
Add	&b
Val	&d

c
Add	&c
Val	&b

*c

*b

d
Add	&d
Val	3

Type	int	int (deref)	int pointer	int pointer (deref)	int double pointer	int
Var Name	a	*b	b	*c	c	d
Point 1	5	5	Add. of a	Add. of a	Add. of b	6
Point 2	8		Add. of d			3
Point 3						

SER 334
Sample Review

```
1   int a, *b, **c, d;
2    a = 5;
3    d = a + 1;
4    b = &a;
5    c = &b
6    // Point 1
7    *b = 8;
8    b = &d;
9    **c = 3;
10   // Point 2
11    b+=7;
12     // Point 3
```
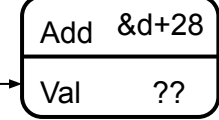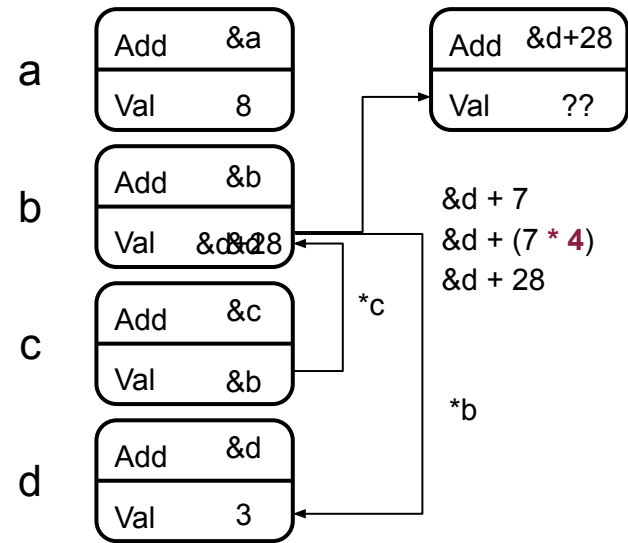
a
```
Add    &a
Val     8
```

```
Add
Val
```

b
```
Add    &b
Val    &d
```

&d + 7
&d + (7 * 4)
&d + 28

*c

c
```
Add    &c
Val    &b
```

*b

d
```
Add    &d
Val     3
```

Type	int	int (deref)	int pointer	int pointer (deref)	int double pointer	int
Var Name	a	*b	b	*c	c	d
Point 1	5	5	Add. of a	Add. of a	Add. of b	6
Point 2	8	3	Add. of d	Add. of d	Add. of b	3
Point 3						

SER 334
Sample Review

```
1   int a, *b, **c, d;
2    a = 5;
3    d = a + 1;
4    b = &a;
5    c = &b
6    // Point 1
7    *b = 8;
8    b = &d;
9    **c = 3;
10   // Point 2
11    b+=7;
12    // Point 3
```

a
| Add | &a |
| Val | 8 |

| Add | &d+28 |
| Val | ?? |

b
| Add | &b |
| Val | &d+28 |

&d + 7
&d + (7 * 4)
&d + 28

c
| Add | &c |
| Val | &b |

*c

d
| Add | &d |
| Val | 3 |

Type	int	int (deref)	int pointer	int pointer (deref)	int double pointer	int
Var Name	a	*b	b	*c	c	d
Point 1	5	5	Add. of a	Add. of a	Add. of b	6
Point 2	8	3	Add. of d	Add. of d	Add. of b	3
Point 3						

SER 334
Sample Review

```
1   int a, *b, **c, d;
2    a = 5;
3    d = a + 1;
4    b = &a;
5    c = &b
6    // Point 1
7    *b = 8;
8    b = &d;
9    **c = 3;
10   // Point 2
11    b+=7;
12    // Point 3
```

a
| Add | &a |
| Val | 8 |

| Add | &d+28 |
| Val | ?? |

b
| Add | &b |
| Val | &d&28 |

&d + 7
&d + (7 * 4)
&d + 28

c
| Add | &c |
| Val | &b |

*c

d
| Add | &d |
| Val | 3 |

*b

Type	int	int (deref)	int pointer	int pointer (deref)	int double pointer	int
Var Name	a	*b	b	*c	c	d
Point 1	5	5	Add. of a	Add. of a	Add. of b	6
Point 2	8	3	Add. of d	Add. of d	Add. of b	3
Point 3		Value @ &d + (7*4)	&d + (7*4)	&d + (7*4)		

SER 334
Practice Exam

```
1    int w = 0, *x = &w, y = 0;
2    // Point 1
3    w = 5;
4    y = -1;
5    // Point 2
6    x = &y;
7    y = 10;
8    w = y + *x;
9    // Point 3
```

w
Add	&w
Val	

x
Add	&x
Val	

y
Add	&y
Val	

Type	int	int (deref)	int pointer	int
Var Name	w	*x	x	y
Point 1				
Point 2				
Point 3				

SER 334
Sample Review

3. [Acuña] Consider the problem of padding the following structure, and answer the three questions below. Assume that you are compiling on a system with a 32-bit architecture. [4 points total]

```
struct bmp_header {
     char creator_name[254];
     int width;
     int height;
     char signature_rgb[2];
     int offset_pixels;
};
```

(a) What is the size of this struct as defined?

(b) How much space would be wasted with word length padding?

(c) [Katie] Redefine the structure to reduce the wasted space.

SER 334

Sample Review

5. [Acuña] Shown below is a valid BMP file that has been opened in a hex editor. Based on the data visible, answer the following questions. Indicate which number base you use for each question. (The complete specification is shown in the appendix.) Note that this screen shot comes from an Intel architecture system where numbers are stored with little-endian byte ordering

```
00000000 42 4D 76 04 00 00 00 00 00 00 36 00 00 00 28 00 00 00 15 00 B
00000014 00 00 11 00 00 00 01 00 18 00 00 00 00 00 40 04 00 00 00 00 .
00000028 00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FF FF FF FF FF .
0000003c FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF .
00000050 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF .
00000064 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00 FF FF .
00000078 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF .
0000008c FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF .
000000a0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF .
000000b4 FF 00 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF .
000000c8 FF FF FF FF FF FF FF FF FF 00 00 00 00 00 00 00 00 00 00 .
000000dc 00 00 00 00 00 00 00 FF FF FF FF FF FF FF FF FF FF FF FF FF .
0000000f FF FF FF FF FF 00 FF FF FF FF FF FF FF FF FF FF FF FF FF FF .
00000104 FF FF FF FF FF FF FF 00 00 00 00 00 00 00 00 00 FF FF FF FF .
00000118 FF FF FF FF FF FF FF FF 00 00 00 00 00 00 00 00 00 FF FF FF .
0000012c FF FF FF FF FF FF FF FF FF 00 FF FF FF FF FF FF FF FF FF FF .
```

2-bytes   2-bytes

Bitmap File Header

Signature

File Size

Reserved1   Reserved2

File Offset to PixelArray

DIB Header

DIB Header Size

Image Width (w)

Image Height (h)

Planes   Bits per Pixel

Compression

Image Size

X Pixels Per Meter

Y Pixels Per Meter

Colors in Color Table

Important Color Count

a)

b)

c)

d)

(a) How large is the file?

(b) What is the width and height of this image?

(c) How many bits per pixel are used?

(d) Using b and c, how many bytes are required for each row? How many are for padding?

SER 334
Practice Exam

2. Consider the following function:

```
int* getNextUserID() {
    int users = db_get_user_count("db1");
    static int uid = users + 1;
    return &uid;
}
```

Will the memory allocation in this function work reliably? [5 points]

(a) Yes - it will return an address.

(b) Yes - the memory used by the return value will persist since it is not stack memory.

(c) No - the memory used will be overridden by any new function call the program makes.

(d) No - it will only work some of the time, depending on when getNextUserID is called.

SER 334
Practice Exam

4. What is the output of the following snippet of code? [5 points]

```c
void update(int *var) {
    int new_int = 10;
    var = &new_int;
}
void main() {
    int num = 20;
    int *pointer = &num;
    update(pointer);
    printf("%d",*pointer);
}
```

(a) 10

(b) 20

(c) A compiler error

(d) A run-time error

6. Consider the following snippet of code from Student.c, which defines the class "Student." Why is the parameter for destroy_student a double pointer? [5 points]

Complete the destroy_student function.

```c
typedef struct student student;

struct student {
    char* name;
    char* major;
    int gpa;
}

void destroy_student(student** s){



}
```

```
struct grade_node {
    int value;
    char assignment[255];
    struct grade_node* next;
};
```

10. The following declaration of a struct is used to represent a node in a linked list of grades. Complete the insert_grades(struct node** head) function. This function needs to read the data from the keyboard and store the values in a new node struct. Then, it adds the new node to the beginning of the linked list. [20 points]

```
void insert_grades(struct node** head) {
```

a
Add	&a
Val	

b
Add	&b
Val	

c
Add	&c
Val	

d
Add	&d
Val	

SI Sessions:

- ~~Sunday, January 28th at 7:00 pm MST~~ **Cancelled - good luck on Exam 1!**
- Monday, January 29th at 7:00 pm MST
- Thursday, February 1st at 7:00 pm MST
- Sunday, February 4th at 7:00 pm MST

Review Sessions:

- Exam 2 Review: TDB
- Exam 3 Review: TBD

Questions?

Survey:
http://bit.ly/ASN2324

Check out our other resources!

tutoring.asu.edu

ASU — Arizona State University

Academic Support Network

Services ˅   Faculty and Staff Resources   About Us ˅   University College

Academic Support

Academic Support Network (ASN) provides a variety of free services in-person and online to help currently enrolled ASU students succeed academically.

Services

Subject Area Tutoring

Need in-person or online help with math, science, business, or engineering courses? Just hop into our Zoom room or drop into a center for small group tutoring. We'll take it from there.

Need help using Zoom?

View the tutoring schedule

View digital resources

Go to Zoom

Writing Tutoring

Need help with undergraduate or graduate writing assignments? Schedule an in-person or online appointment, access your appointment link, or wait in our drop-in queue.

Access your appointment link

Access the drop-in queue

Schedule Appointment

Online Study Hub

Join our online peer communities to connect with your fellow Sun Devils. Engage with our tools to search our bank of resources, videos, and previously asked questions. Or, ask our Tutorbot questions.

Now supporting courses in Math, Science, Business, Engineering, and Writing.

Online Study Hub

1 – **Go to Zoom**

Need help using Zoom?

2 – View the tutoring schedule

View digital resources

1. **Click on 'Go to Zoom' to log onto our Online Tutoring Center.**
2. **Click on 'View the tutoring schedule' to see when tutors are available for specific courses.**

More Questions?
Check out our other resources!

tutoring.asu.edu/online-study-hub

Don't forget to check out
the Online Study Hub
for additional resources!

Additional Resources

- **Course Repo**
- **Course Discord**
- **BMP File Format (Wiki)**
- **Linux Kernel API**