



CS 161

Intro to CS I

Finish Recursion/Begin Memory Model

Odds and Ends

- Assignment 4 demo
- Peer Reviews Thursday

Dynamic Memory Demo...



Oregon State University
College of Engineering



What is an Array?

- **Array (ar-ray) *n.*** An ordered arrangement of related items.
 - Example: Array of colors in a rainbow.
 - Related items?
 - Ordered arrangement?
 - Class examples?
 - Computer Science
 - Same data type/data structure
 - Contiguous memory locations

Create 1-D Array



```
int student_grades[5];
```



- How do you access each item?
- What does the array name represent?
- Why is the array name the address of 1st element?
- What are the initial values?



Initialize/Assign Values

- **Declaration**

```
int student_grades[5] = {0, 0, 0, 0, 0};
```

- **Individual Elements**

```
student_grades[0]=0;
```

...

```
student_grades[4]=0;
```

- **Why is this incorrect?**

```
student_grades={0, 0, 0, 0, 0};
```



Initialize/Assign Values...

- **Using a Loop**

- While Loop Example:**

- ```
i=0;
while (i<5) {
 student_grades[i]=0;
 i++;
}
```

- For Loop Example:**

- ```
for(i=0; i<5; i++)
    student_grades[i]=0;
```

- Which is better to use with arrays and why?



Read/Print 1-D Array Values

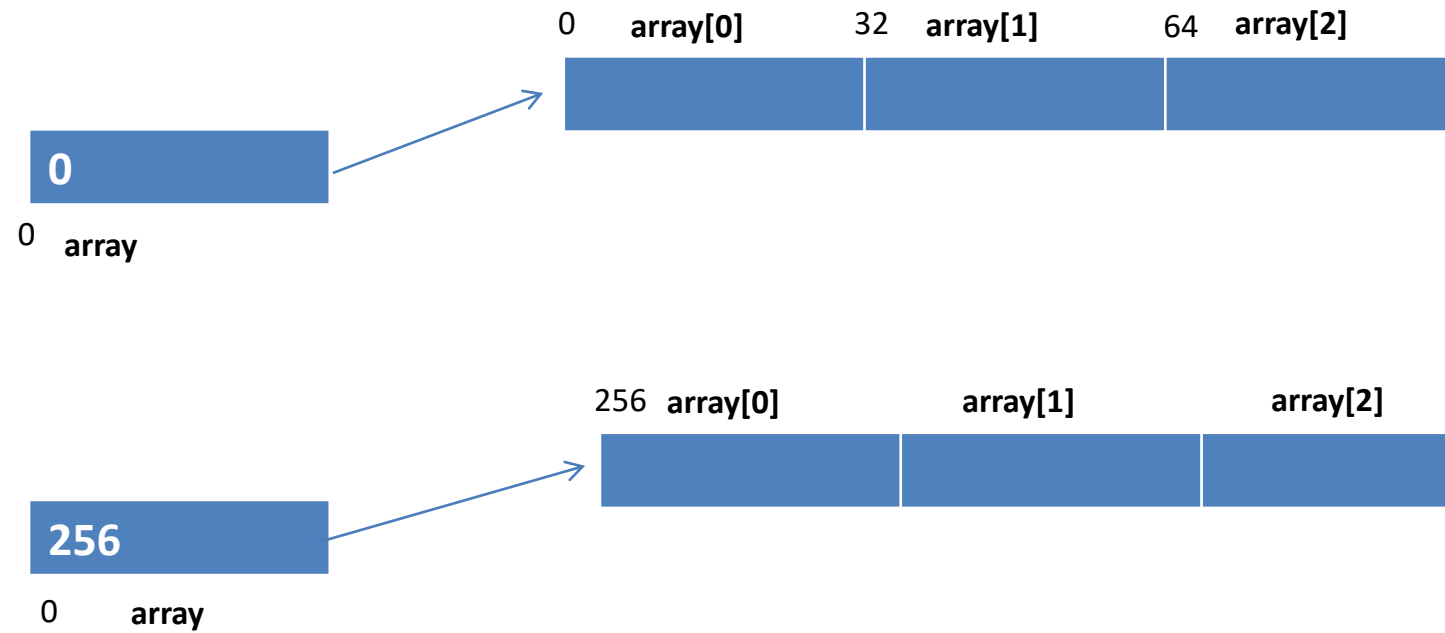
- Read Values From User

```
for(i=0; i<5; i++) {  
    cout << "Enter final grade for student: ";  
    cin >> student_grades[i];  
}
```

- Print Values

```
for (i=0; i<5; i++) {  
    cout << "Student\'s final grade is " << student_grades[i] << endl;  
}
```


Static vs. Dynamic 1-D arrays...



C-Style Strings Demo...



Oregon State University
College of Engineering