DEFAULT PARAMETERS, OPERATOR OVERLOADING

Problem Solving with Computers-II

https://ucsb-cs24-sp17.github.io/

Read the syllabus. Know what's required. Know how to get help.

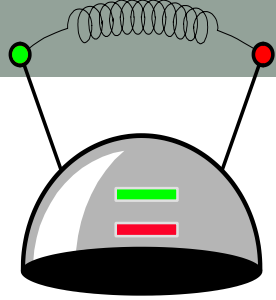
CLICKERS OUT – FREQUENCY AB

Attention all female students

- Women in Computer Science (WICS) is holding a special coffee hour themed “HOW TO SUCCEED IN CS?”
- When? Friday (04/21) at 1:30pm
- Where? HFH 1132

- Please plan to attend! RSVP via this form: https://goo.gl/forms/yPlMyFUN7mWx0vCs1

Review: Constructor



Which constructor is called when the following statement is executed?

```
thinking_cap student;
```

```
class thinking_cap
{
public:
    thinking_cap();                                          //A
    thinking_cap(char new_green[], char new_red[]); //B
    void slots(char new_green[ ], char new_red[ ]);
    void push_green( ) const;
    void push_red( ) const;
private:
    char green_string[50];
    char red_string[50];
};
```

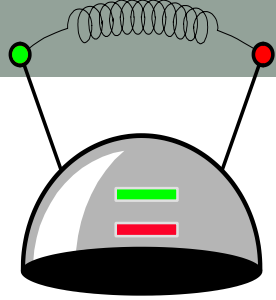
//C: Default copy constructor
//D: Default assignment operator
//E: None of the above

Default values

```
int sum(int a=10, int b=20){
        return a+b;
}

int main(){
        int x= 40, y=50;
        cout<<sum(x,y)<<endl;
        cout<<sum(x)<<endl;
        cout<<sum()<<endl;
}
```

Specify default constructor using default arguments



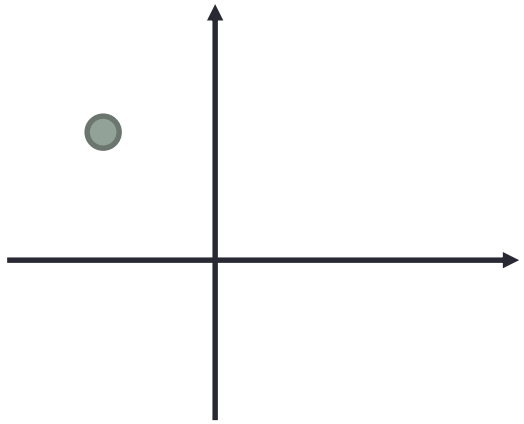
Which constructor is called when the following statement is executed?

```
thinking_cap student;
```

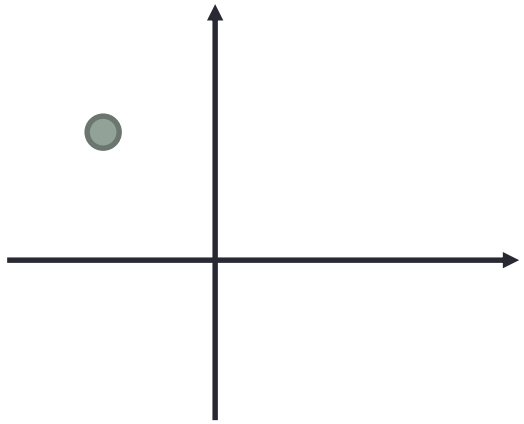
```
class thinking_cap
{
public:
    thinking_cap(char new_green[]="Hello", char new_red[]="there"); //A
    void slots(char new_green[ ], char new_red[ ]);
    void push_green( ) const;
    void push_red( ) const;
private:
    char green_string[50];
    char red_string[50];
};
```

//B: Default copy constructor
//C: Default assignment operator
//D: None of the above

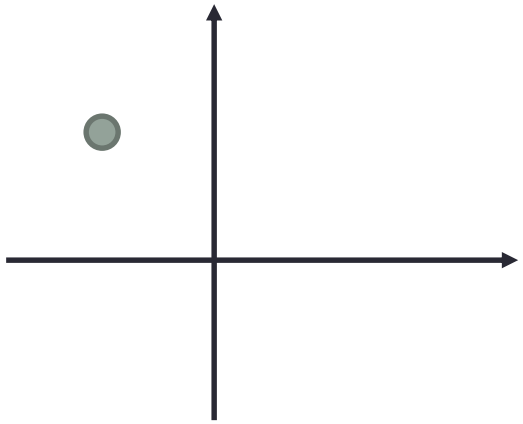
The point class (Chapter 2, section 2.4)



point: (x,y)



shift(delx, dely)



rotate90()

Let’s look at the implementation of the point class

See code from lecture

- Focus on the process : test driven
- Point class definition and implementation
- Operator overloading

Next time

- Wrap up chapter 2, gdb