

9723025

مصطفی حمیدی فرد

گزارش کار تمرین دوم:

پاسخ سوال 1:

```
const Room r2(2, 35, 50);
```

برای اجرای این خط نیاز به یک constructor داریم که 3 ورودی بگیرد. همچنین چون const هست ، باید توابعی که استفاده می کنیم دارای const باشد تا اجرا شوند.

```
r2.changeHumidityStdev(83);
```

باید تابع changeHumidityStdev را تعریف کنیم و قابلیت const داشته باشد و در آن با استفاده از پونتر dht11 انحراف معیار شیء درست شده از dht11 را تغییر دهیم پس باید هنگام construct یک dht11 هم بسازیم. همچنین در dht11 باید تابع set برای تغییر داشته باشیم.

```
std::pair<double, double> p{r2.  
dht->getStdevs()};
```

برای اجرای این برنامه نیاز به یک تابع داریم با نام getStdevs که در کلاس dht11 تعریف شده است و خروجی آن از جنس std::pair<double , double> است. و در آن مقادیر انحراف معیار را برگرداند.

پاسخ سوال 2:

نمی توان std::unique_ptr را کپی کرد و چون هنگام صدا زدن function مقدار آن کپی می شود پس unique_ptr را نمی توان از این طریق در تابع صدا زد و باید دقیقا خود آن را (مثلا از طریق رفرنس) صدا زد.

توضیحاتی در مورد کد:

```
class Room; // forward declaration for using cyclo dependant classes
```

چون دو کلاس مورد استفاده در هم تنیده هستند ، نمی توان به صورت عادی از include استفاده کرد و باید از forward declaration استفاده کرد تا مشکلات مورد نظر رفع شود و پوینتر در کلاس dht11 ساخته شود.

```
Room::Room(size_t _id, double _temperature, double _humidity)  
: id{_id}, temperature{_temperature}, humidity{_humidity}, dht{new DHT11{0, 0, this}} {}
```

نکته بعدی آن است که در برای مقدار اولیه دادن به dht که از جنس unique_ptr هست ، نمی توان از عملگر = کمک گرفت و به خاطر همین ، آن را به صورت بالا تعریف کرده ام.

```
double DHT11::getTemperature() const  
{  
    unsigned seed = std::chrono::system_clock::now().time_since_epoch().count();  
    std::default_random_engine generator(seed);  
    std::normal_distribution<double> distribution(proom->getTemperature(), temp_stdev);  
    return distribution(generator);  
}
```

این چند خط برای ایجاد متغیر تصادفی گاوسی هست. خط اول برای آن هست که seed براساس زمان ، هر دفعه تغییر کند و با استفاده از دو خط بعدی می توان یک تابع گاوسی ساخت و تابع موجود در خط آخر نیز یک عدد اعشاری تصادفی از این متغیر تصادفی برمی گرداند.

تابع هایی به کلاس ها اضافه شده اند که در تمرین استفاده نشده اند ولی در واقعیت اگر قرار بود یک چنین کلاسی ساخته شود ، باید وجود می داشتند در نتیجه من آن ها را در یک حرکت خودجوش 😊 اضافه کرده ام.(مثل تابعی که id اتاق ها را نشان دهد یا عوض کند)

در کل به نظر خودم بقیه کد چالش خاصی ندارد و سعی کرده ام جاهایی که واقعا برایم چالش بوده است را در اینجا توضیح بدهم.