فاز دوم پروژه برنامه نویسی پیشرفته

مديريت بيمارستان

مدرس دکتر نرگس السادات بطحاییان

> عرفان يوسفى آژنگ 9912358044

#### مقدمه

در این فاز ارث بری به کلاس ها اضافه شد و جلوگیری از تعریف بعضی از کلاس ها و خطوط متعدد شد.

# شرح فاز

در این فاز کلاس بیمارستان به پروژه اضافه شد.

کلاس بیمار و کلاس دکتر از کلاس شخص ارث بردند و پیاده سازی شدند.

### Class Person

```
class Person
     public:
          enum Gender{Male, Female};
          Person(std::string name , unsigned int age , Gender gender , std::string phone);
          //Destructor
          virtual ~Person();
          void set_name(std::string);
          void set_age(unsigned int);
          void set_gender(Gender);
          void set_phone(std::string);
          std::string get_name() const;
          unsigned int get_age() const;
          Gender get_gender() const;
          std::string get_phone() const;
          virtual void show_inf() const;
     private:
          std::string name;
          unsigned int age;
          Gender gender;
          std::string phone_number;
};
```

### Class Doctor: Public Person

# این کلاس از کلاس شخص ارث برده است

```
#include "Person.hpp"
#include <string>
#include <vector>
class Doctor : public Person
     public:
         enum Specialization {General, Expert, Surgeon};
          Doctor(std::string name, unsigned int age, Gender gender, std::string phone_number, Specialization specialization);
          virtual ~Doctor();
         void make_id();
          void set_id(unsigned int);
          void set_specialization(Specialization);
          unsigned int get_id() const;
          Specialization get_specialization() const;
          virtual void show_inf() const;
    private:
          unsigned int id; // id
          Specialization specialization = General;// takhasos
          static std::vector<unsigned int> id_list;
#endif
```

### Class Patient: Public Person

این کلاس از کلاس شخص ارث برده است

```
class Patient: public Person

public:

//enum

enum Blood_types {A, B, AB, O, Unknown};

enum Insurance {No, Yes};

enum Status {Nonen, Yes};

//Constructor

patient();

//Sestructor

virtual "Patient();

//set and get functions

void set_discod_type(Blood_types);

void set_discod_type(Blood_types);

void set_discod_type(Blood_type);

void set_dissurance(Insurance);

Status get_status() const;

Blood_types get_blood_type() const;

Illness get_filness() const;

Insurance get_insurance() const;

// show information

virtual void show_inf() const;

private:

//Dota members

Status status - Naiting: // varlyat martz

Blood_types blood_type = Unknown; // garoh bhooni

Illness illness = Noknown_Illness; // bisser

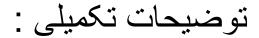
Insurance insurance = No; // bisse

Insurance insurance = No; // bisse

Insurance insurance = No; // bisse
```

## **Class Hospital**

```
#ifndef HOSPITAL_HPP
   #define HOSPITAL_HPP
3
   #include <Doctor.hpp>
   #include <Patient.hpp>
   #include <vector>
   #include <string>
   class Hospital
         public:
              Hospital();
              virtual ~Hospital();
              void add_doctor(Doctor);
              void add_patient(Patient);
              void set_visit(std::string, std::string);
              unsigned int set_room_for_visit();
              Doctor & search_doctor(std::string);
              Patient & search_patient(std::string);
              virtual void show_hospital_information() const;
         private:
              std::vector<Doctor> doctors;
              std::vector<Patient> patients;
              static std::vector<unsigned int> rooms;
    };
    #endif
```



پیاده سازی ها به صورت ساده و موقت نوشته شده و در فاز بعدی اکسپشن اضافه میشود.

عرفان يوسفي آژنگ

9912358044