Цэрэнбат Номин-Эрдэнэ

(20B1NUM0426)

ICSI201 - Объект хандлагат програмчлал

МУИС-ХШУИС

Мэдээллийн технологи

Лаборатори №8

Цэрэнбат Номин-Эрдэнэ

1. ОРШИЛ

Энэ лабораторийн ажлийн хүрээнд Обьект хандлагат С хэлний чухал хэсэг болох удамшил ямар харьцаа үүсгэдэг тухай болон бүрдэл харьцааг судалж, тайлбарлаж, ойлгож авна.

2. ЗОРИЛГО

- Лабораторийн ажлыг алдаагүй зөв хийж гүйцэтгэх.
- Division болон JobDescription классуудын хэд хэдэн объект байгуулах
- Employee классын хэд хэдэн объект байгуулж тус бүрийн Division, JobDescription –ийг тодорхойлох.
- Employee классын объект тус бүрд Spouse, Child – уудыг тохируулж өгөх
- Employee классын объект тус бүрийн бүх мэдээллийг хэвлэх.

3. ОНОЛЫН СУДАЛГАА

3.1 Удамшил ямар харьцаа үүсгэдэг вэ? Объект хандлагат программчлалд хэрхэн хэрэгжүүлдэг вэ?

Удамших болон удамшуулах харьцаанд орж буй классуудын хооронд тэр бол тэр эсвэл тийм төрлийх гэсэн хоёр харьцаа үүсгэдэг. Жишээлвэл тэр бол тэр гэдэг нь хүүхэд бол хүн харин тийм төрлийх гэдэгт нь машин бол тээврийн хэрэгсэлийн төрлийх гэх мэтчилэн.

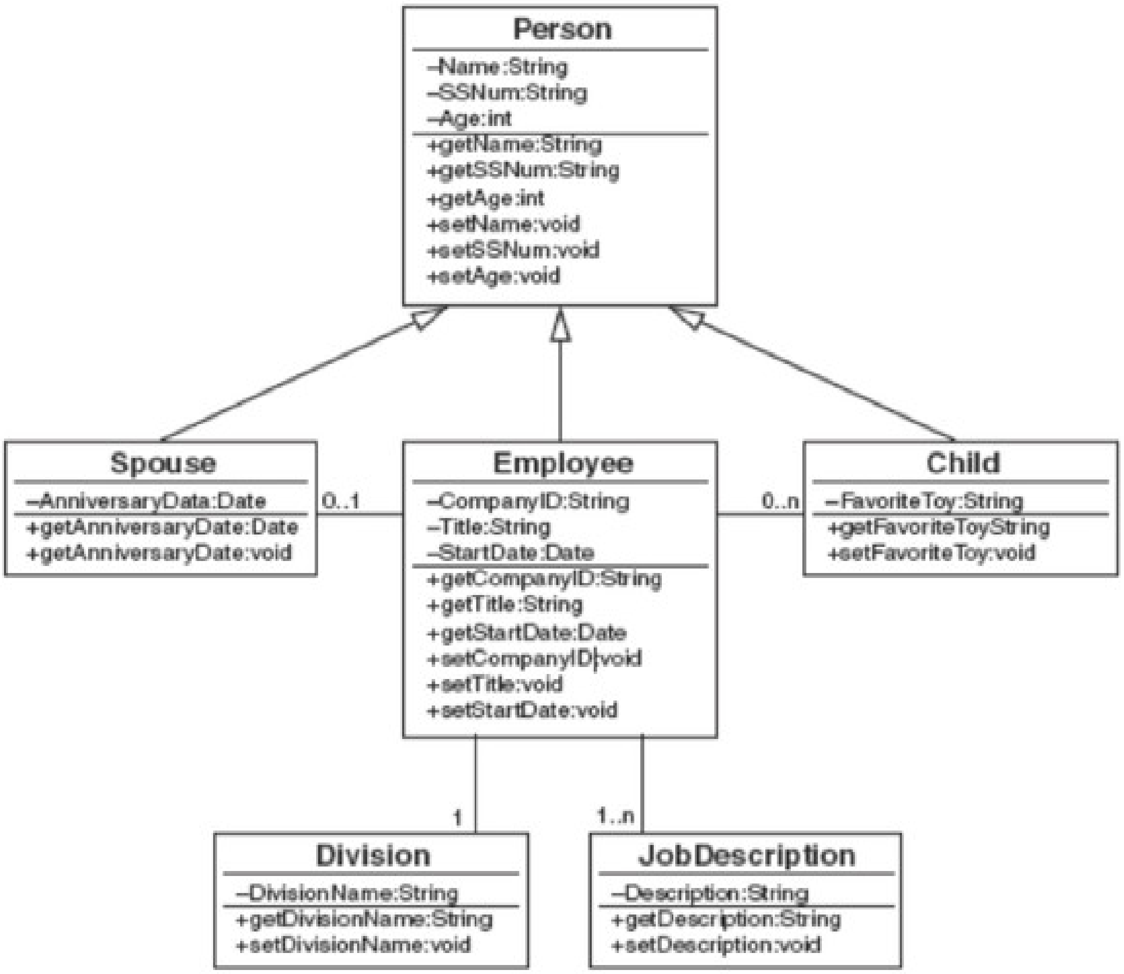
3.2 Бүрдэл харьцаа гэж юу вэ? Объект хандлагат программчлалд хэрхэн хэрэгжүүлдэг вэ?

Класын шинж нь өөр класых байж болох ба нарийн бүтэцтэй бүрдмэл классыг үүсгэхэд ашигладаг. Энэ нь тийм юмтай байх гэсэн харьцааг үсгэдэг.

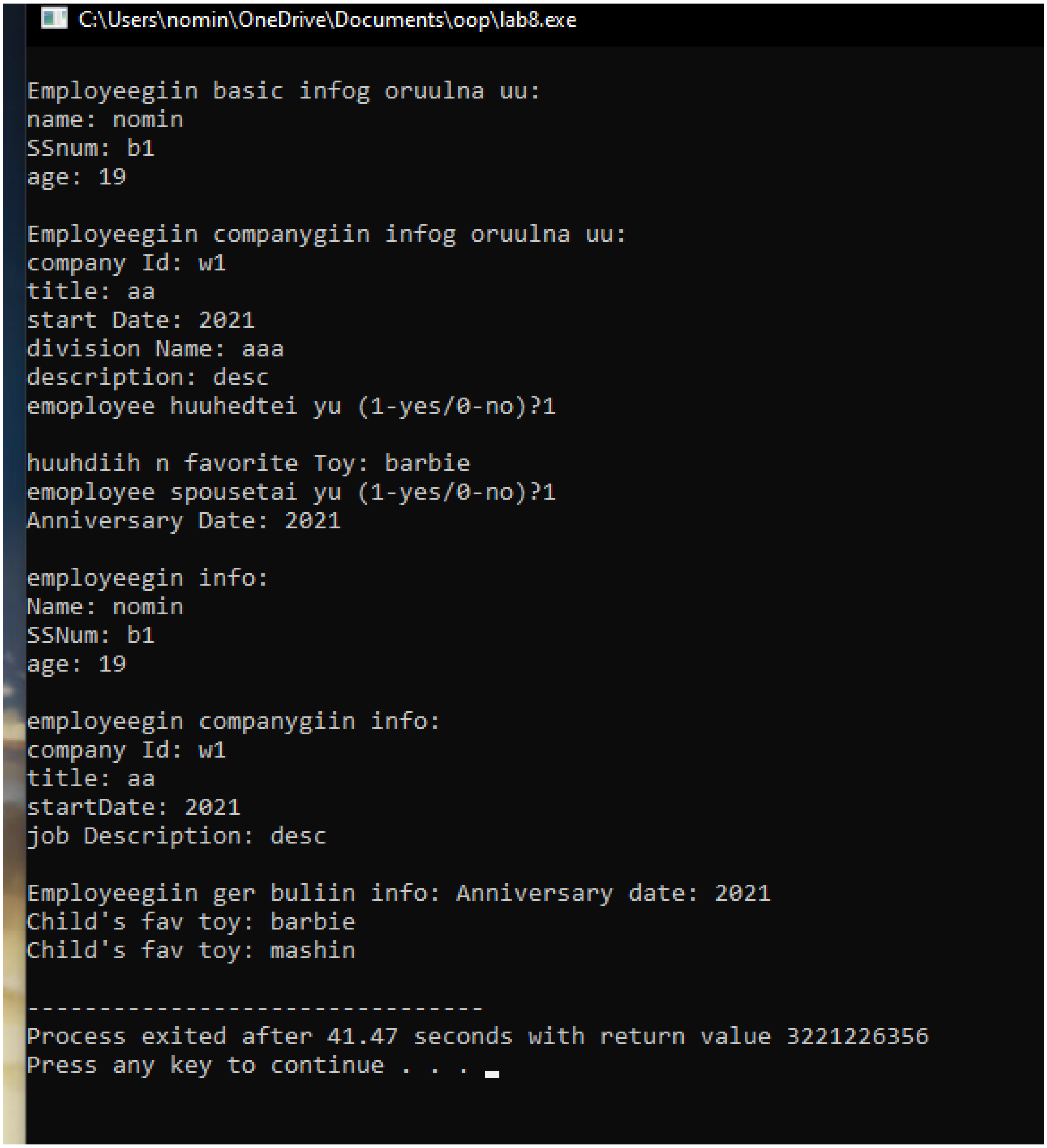
```
Class engine{
	Public:int c;
};
Class car {
	Public: char name[15];
	engine e1; //mashin ni motortoi baix estoi
	engine e2;
};
```

4. ХЭРЭГЖҮҮЛЭЛТ

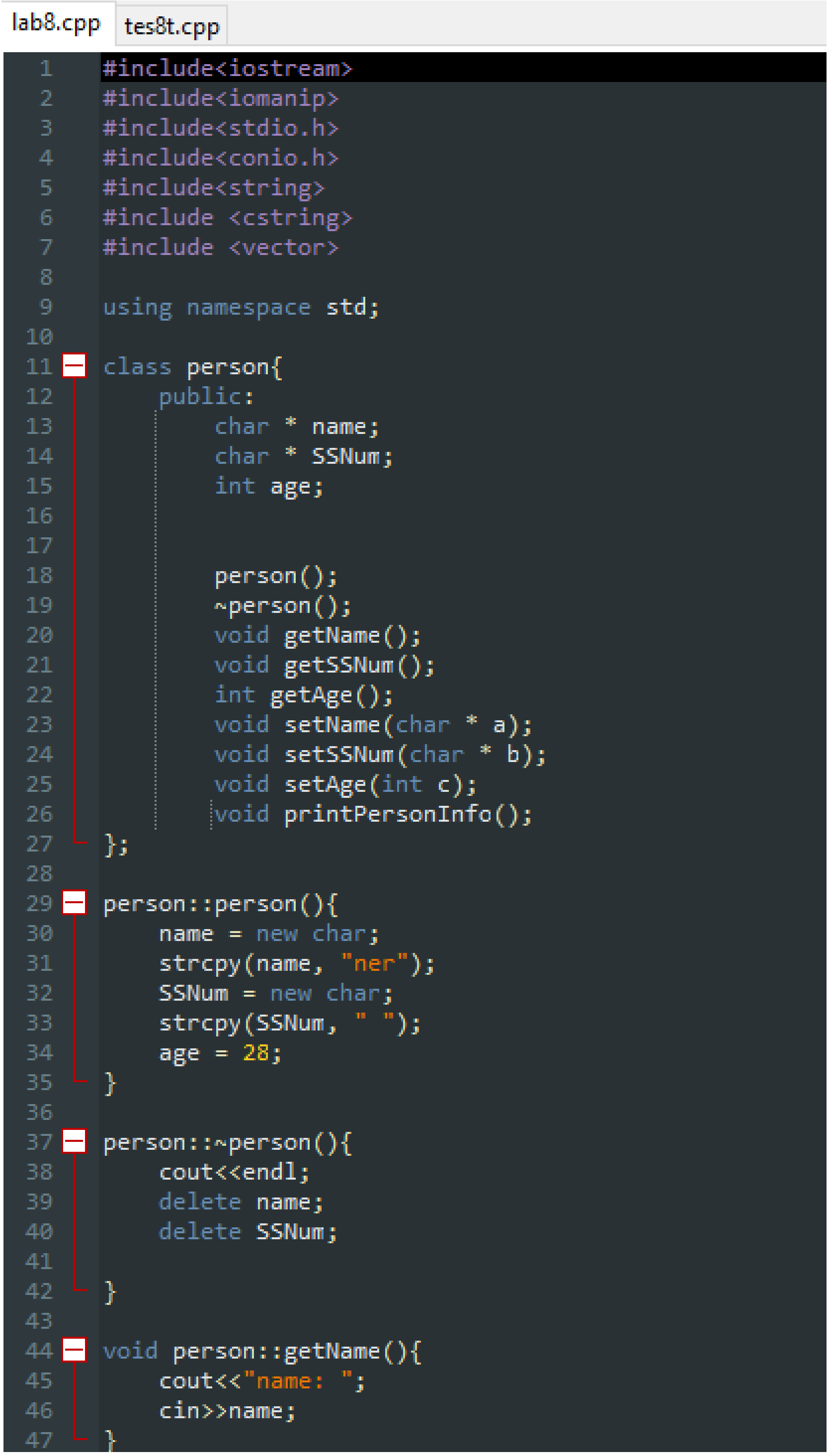
1. Дараах зурагт харуулсан класс диаграмын дагуу классуудаг байгуул.
2. Division болон JobDescription классуудын хэд хэдэн объект байгуул
3. Employee классын хэд хэдэн объект байгуулж тус бүрийн Division, JobDescription –ийг зааж өг.
4. Employee классын объект тус бүрд Spouse, Child – уудыг тохируулж өг
5. Employee классын объект тус бүрийн бүх мэдээллийг хэвлэ.
6. 0..1, 0..n болон 1, 1..n харьцаануудыг зөв програмчилсан байх шаардлагатай.



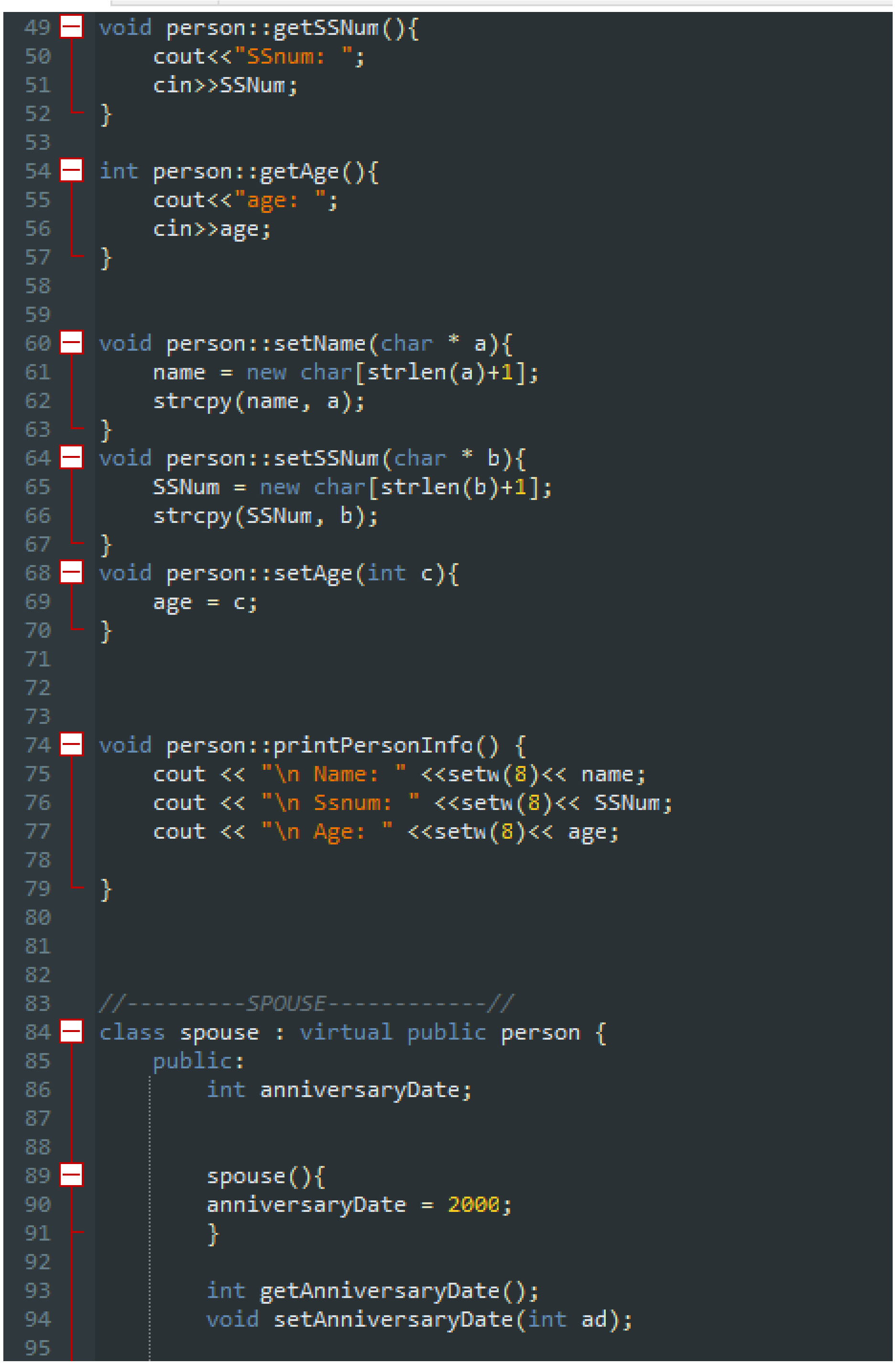
Cardinality in a UML diagram.



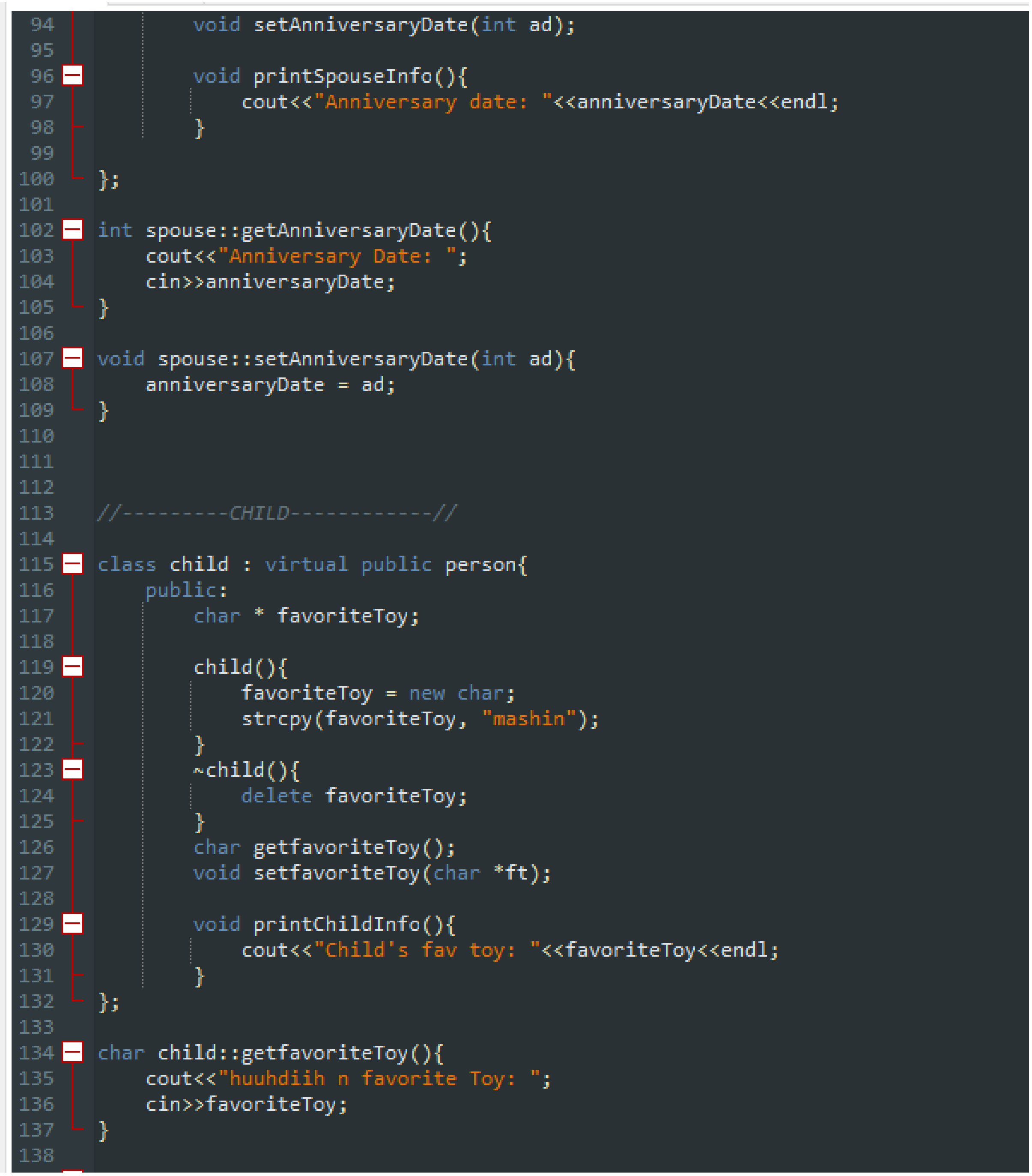
```
Employeegiin basic infog oruulna uu:
name: nomin
SSnum: b1
age: 19

Employeegiin companygiin infog oruulna uu:
company Id: w1
title: aa
start Date: 2021
division Name: aaa
description: desc
emoployee huuhedtei yu (1-yes/0-no)?1

huuhdiih n favorite Toy: barbie
emoployee spousetai yu (1-yes/0-no)?1
Anniversary Date: 2021

employeegin info:
Name: nomin
SSNum: b1
age: 19

employeegin companygiin info:
company Id: w1
title: aa
startDate: 2021
job Description: desc

Employeegiin ger buliin info: Anniversary date: 2021
Child's fav toy: barbie
Child's fav toy: mashin

--------------------------------
Process exited after 41.47 seconds with return value 3221226356
Press any key to continue . . .
```



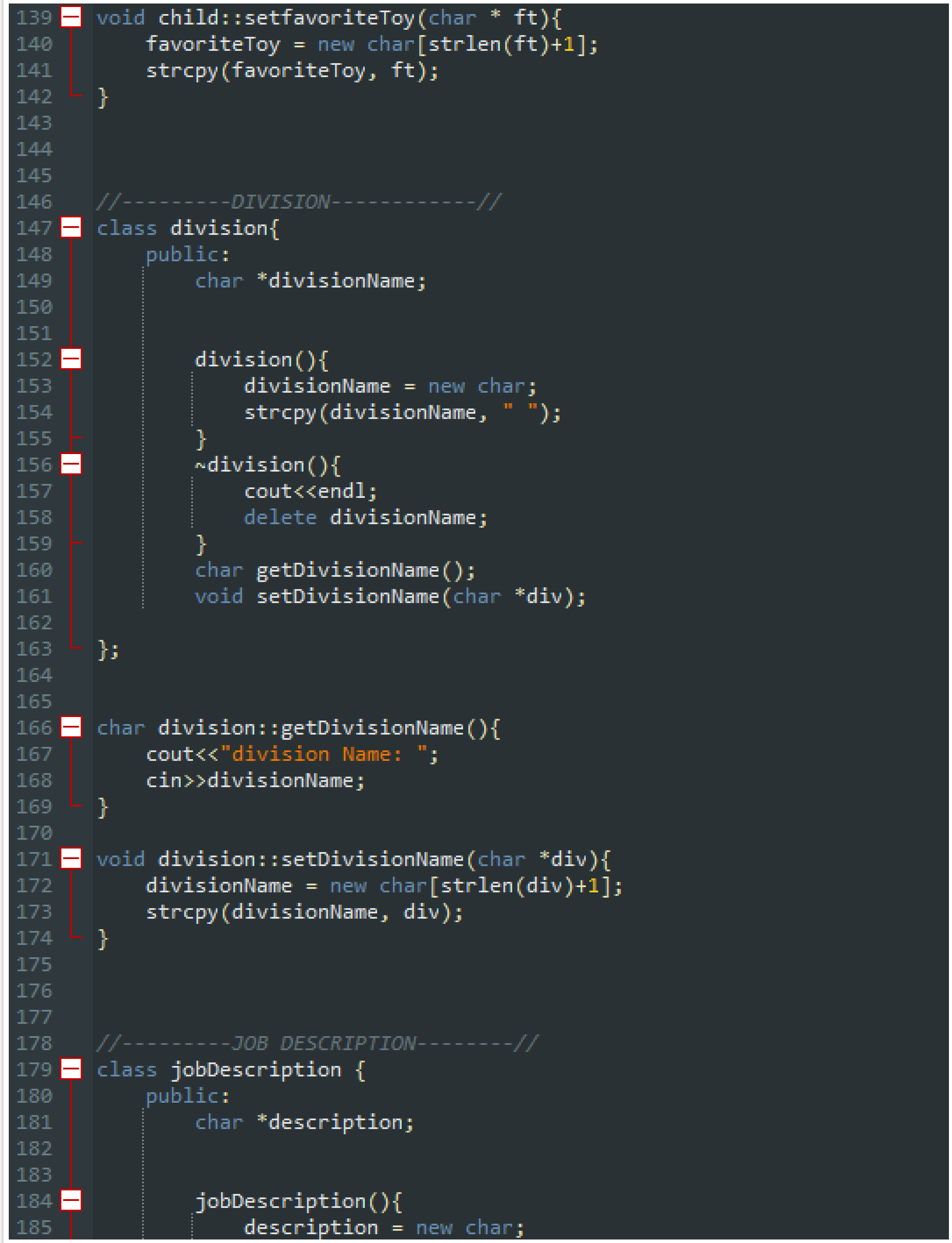
```cpp
#include<iostream>
#include<iomanip>
#include<stdio.h>
#include<conio.h>
#include<string>
#include <cstring>
#include <vector>

using namespace std;

class person{
    public:
        char * name;
        char * SSNum;
        int age;


        person();
        ~person();
        void getName();
        void getSSNum();
        int getAge();
        void setName(char * a);
        void setSSNum(char * b);
        void setAge(int c);
        void printPersonInfo();
};

person::person(){
    name = new char;
    strcpy(name, "ner");
    SSNum = new char;
    strcpy(SSNum, " ");
    age = 28;
}

person::~person(){
    cout<<endl;
    delete name;
    delete SSNum;

}

void person::getName(){
    cout<<"name: ";
    cin>>name;
}
```



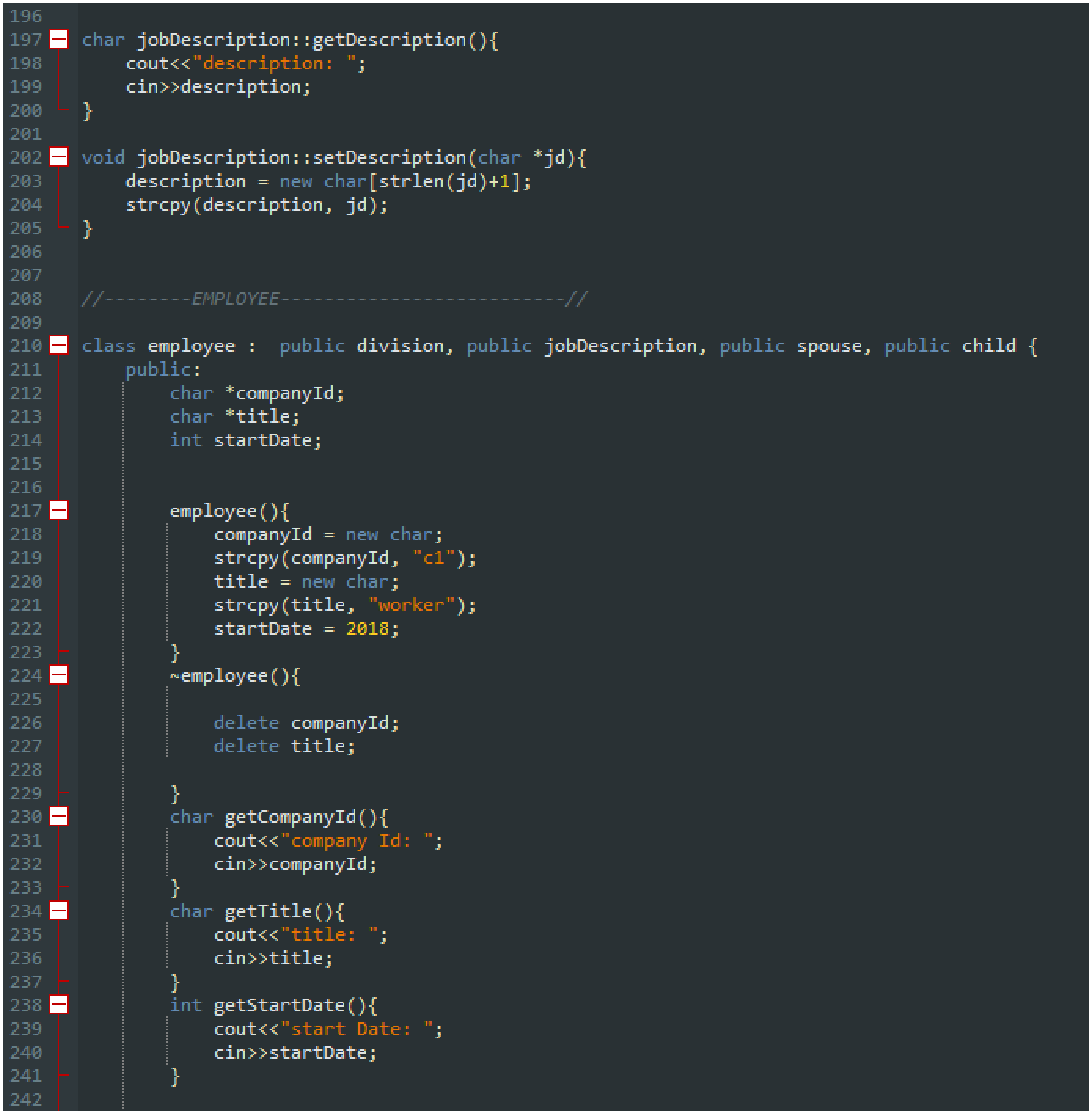
```cpp
void person::getSSNum(){
    cout<<"SSnum: ";
    cin>>SSNum;
}

int person::getAge(){
    cout<<"age: ";
    cin>>age;
}


void person::setName(char * a){
    name = new char[strlen(a)+1];
    strcpy(name, a);
}
void person::setSSNum(char * b){
    SSNum = new char[strlen(b)+1];
    strcpy(SSNum, b);
}
void person::setAge(int c){
    age = c;
}



void person::printPersonInfo() {
    cout << "\n Name: " <<setw(8)<< name;
    cout << "\n Ssnum: " <<setw(8)<< SSNum;
    cout << "\n Age: " <<setw(8)<< age;

}



//---------SPOUSE------------//
class spouse : virtual public person {
    public:
        int anniversaryDate;


        spouse(){
        anniversaryDate = 2000;
        }

        int getAnniversaryDate();
        void setAnniversaryDate(int ad);

```



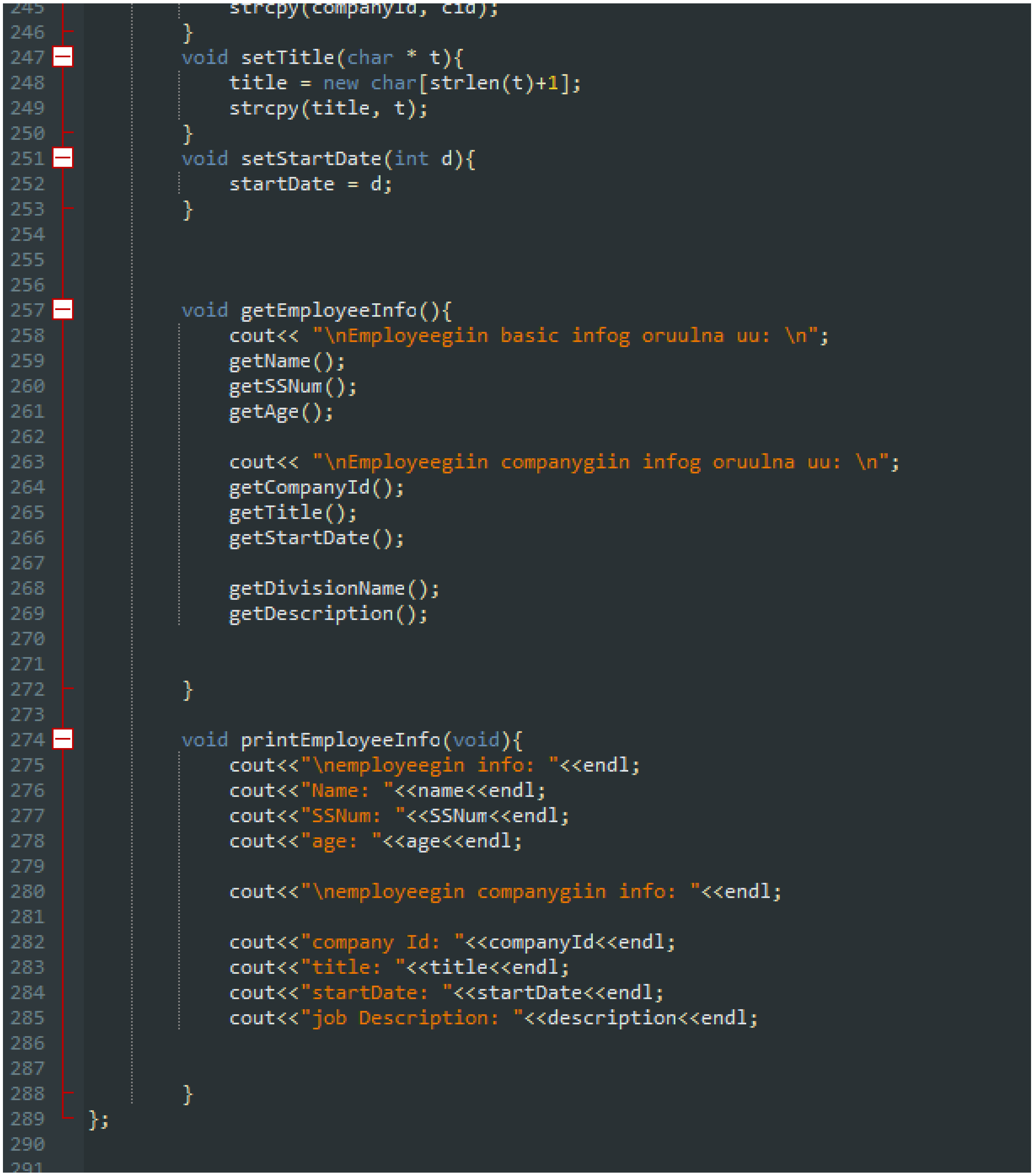
```
        void setAnniversaryDate(int ad);

        void printSpouseInfo(){
            cout<<"Anniversary date: "<<anniversaryDate<<endl;
        }

};

int spouse::getAnniversaryDate(){
    cout<<"Anniversary Date: ";
    cin>>anniversaryDate;
}

void spouse::setAnniversaryDate(int ad){
    anniversaryDate = ad;
}



//---------CHILD-----------//

class child : virtual public person{
    public:
        char * favoriteToy;

        child(){
            favoriteToy = new char;
            strcpy(favoriteToy, "mashin");
        }
        ~child(){
            delete favoriteToy;
        }
        char getfavoriteToy();
        void setfavoriteToy(char *ft);

        void printChildInfo(){
            cout<<"Child's fav toy: "<<favoriteToy<<endl;
        }
};

char child::getfavoriteToy(){
    cout<<"huuhdiih n favorite Toy: ";
    cin>>favoriteToy;
}

```



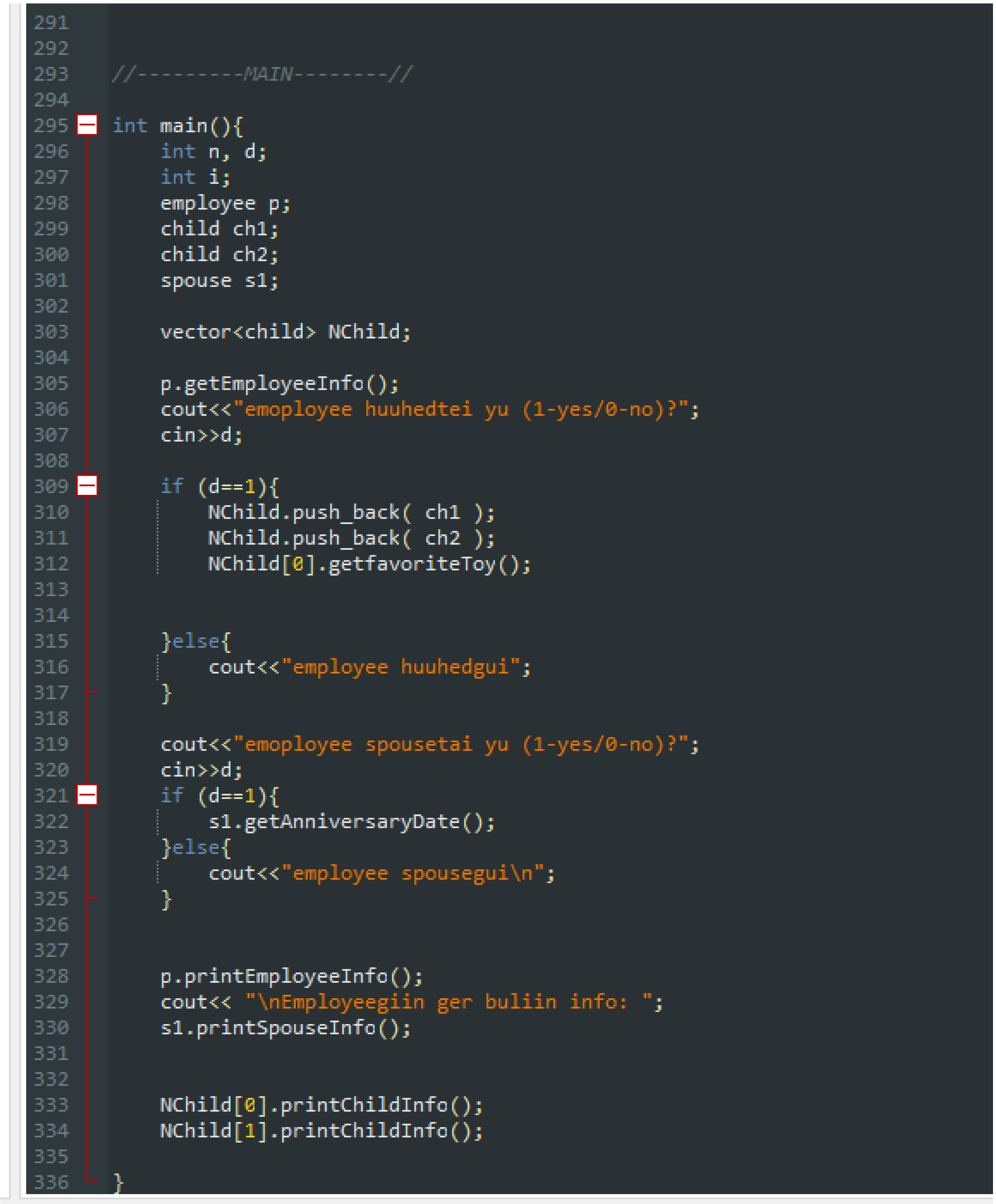
```cpp
void child::setfavoriteToy(char * ft){
    favoriteToy = new char[strlen(ft)+1];
    strcpy(favoriteToy, ft);
}



//---------DIVISION-----------//
class division{
    public:
        char *divisionName;


        division(){
            divisionName = new char;
            strcpy(divisionName, " ");
        }
        ~division(){
            cout<<endl;
            delete divisionName;
        }
        char getDivisionName();
        void setDivisionName(char *div);

};


char division::getDivisionName(){
    cout<<"division Name: ";
    cin>>divisionName;
}

void division::setDivisionName(char *div){
    divisionName = new char[strlen(div)+1];
    strcpy(divisionName, div);
}



//---------JOB DESCRIPTION--------//
class jobDescription {
    public:
        char *description;


        jobDescription(){
            description = new char;
```



```

char jobDescription::getDescription(){
    cout<<"description: ";
    cin>>description;
}

void jobDescription::setDescription(char *jd){
    description = new char[strlen(jd)+1];
    strcpy(description, jd);
}


//--------EMPLOYEE-------------------------//

class employee :  public division, public jobDescription, public spouse, public child {
    public:
        char *companyId;
        char *title;
        int startDate;


        employee(){
            companyId = new char;
            strcpy(companyId, "c1");
            title = new char;
            strcpy(title, "worker");
            startDate = 2018;
        }
        ~employee(){

            delete companyId;
            delete title;

        }
        char getCompanyId(){
            cout<<"company Id: ";
            cin>>companyId;
        }
        char getTitle(){
            cout<<"title: ";
            cin>>title;
        }
        int getStartDate(){
            cout<<"start Date: ";
            cin>>startDate;
        }

```



```
            strcpy(companyId, cId);
        }
        void setTitle(char * t){
            title = new char[strlen(t)+1];
            strcpy(title, t);
        }
        void setStartDate(int d){
            startDate = d;
        }



        void getEmployeeInfo(){
            cout<< "\nEmployeegiin basic infog oruulna uu: \n";
            getName();
            getSSNum();
            getAge();

            cout<< "\nEmployeegiin companygiin infog oruulna uu: \n";
            getCompanyId();
            getTitle();
            getStartDate();

            getDivisionName();
            getDescription();


        }

        void printEmployeeInfo(void){
            cout<<"\nemployeegin info: "<<endl;
            cout<<"Name: "<<name<<endl;
            cout<<"SSNum: "<<SSNum<<endl;
            cout<<"age: "<<age<<endl;

            cout<<"\nemployeegin companygiin info: "<<endl;

            cout<<"company Id: "<<companyId<<endl;
            cout<<"title: "<<title<<endl;
            cout<<"startDate: "<<startDate<<endl;
            cout<<"job Description: "<<description<<endl;


        }
};
```



```
//---------MAIN--------//

int main(){
    int n, d;
    int i;
    employee p;
    child ch1;
    child ch2;
    spouse s1;

    vector<child> NChild;

    p.getEmployeeInfo();
    cout<<"emoployee huuhedtei yu (1-yes/0-no)?";
    cin>>d;

    if (d==1){
        NChild.push_back( ch1 );
        NChild.push_back( ch2 );
        NChild[0].getfavoriteToy();


    }else{
        cout<<"employee huuhedgui";
    }

    cout<<"emoployee spousetai yu (1-yes/0-no)?";
    cin>>d;
    if (d==1){
        s1.getAnniversaryDate();
    }else{
        cout<<"employee spousegui\n";
    }


    p.printEmployeeInfo();
    cout<< "\nEmployeegiin ger buliin info: ";
    s1.printSpouseInfo();


    NChild[0].printChildInfo();
    NChild[1].printChildInfo();

}
```

5. ХАВСРАЛТ

```
#include<iostream>
#include<iomanip>
#include<stdio.h>
#include<conio.h>
#include<string>
```

```
#include <cstring>
#include <vector>

using namespace std;

class person{
        public:
                char * name;
                char * SSNum;
                int age;


                person();
                ~person();
                void getName();
                void getSSNum();
                int getAge();
                void setName(char * a);
                void setSSNum(char * b);
                void setAge(int c);
                void printPersonInfo();
};

person::person(){
        name = new char;
        strcpy(name, "ner");
        SSNum = new char;
        strcpy(SSNum, " ");
        age = 28;
}

person::~person(){
        cout<<endl;
        delete name;
        delete SSNum;

}

void person::getName(){
        cout<<"name: ";
        cin>>name;
}

void person::getSSNum(){
        cout<<"SSnum: ";
        cin>>SSNum;
```

```
}

int person::getAge(){
    cout<<"age: ";
    cin>>age;
}


void person::setName(char * a){
    name = new char[strlen(a)+1];
    strcpy(name, a);
}
void person::setSSNum(char * b){
    SSNum = new char[strlen(b)+1];
    strcpy(SSNum, b);
}
void person::setAge(int c){
    age = c;
}



void person::printPersonInfo() {
    cout << "\n Name: " <<setw(8)<< name;
    cout << "\n Ssnum: " <<setw(8)<< SSNum;
    cout << "\n Age: " <<setw(8)<< age;

}



//---------SPOUSE------------//
class spouse : virtual public person {
    public:
        int anniversaryDate;


        spouse(){
        anniversaryDate = 2000;
        }

        int getAnniversaryDate();
        void setAnniversaryDate(int ad);

        void printSpouseInfo(){
            cout<<"Anniversary date: "<<anniversaryDate<<endl;
        }

};

int spouse::getAnniversaryDate(){
    cout<<"Anniversary Date: ";
    cin>>anniversaryDate;
}

void spouse::setAnniversaryDate(int ad){
    anniversaryDate = ad;
}



//---------CHILD------------//

class child : virtual public person{
    public:
        char * favoriteToy;

        child(){
            favoriteToy = new char;
            strcpy(favoriteToy, "mashin");
        }
        ~child(){
            delete favoriteToy;
        }
        char getfavoriteToy();
        void setfavoriteToy(char *ft);

        void printChildInfo(){
            cout<<"Child's fav toy: "<<favoriteToy<<endl;
        }
};

char child::getfavoriteToy(){
    cout<<"huuhdiih n favorite Toy: ";
    cin>>favoriteToy;
}

void child::setfavoriteToy(char * ft){
    favoriteToy = new char[strlen(ft)+1];
    strcpy(favoriteToy, ft);
}
```

```
//---------DIVISION------------//
class division{
        public:
                char *divisionName;


                division(){
                        divisionName = new char;
                        strcpy(divisionName, " ");
                }
                ~division(){
                        cout<<endl;
                        delete divisionName;
                }
                char getDivisionName();
                void setDivisionName(char *div);

};


char division::getDivisionName(){
        cout<<"division Name: ";
        cin>>divisionName;
}

void division::setDivisionName(char *div){
        divisionName = new char[strlen(div)+1];
        strcpy(divisionName, div);
}



//---------JOB DESCRIPTION--------//
class jobDescription {
        public:
                char *description;


                jobDescription(){
                        description = new char;
                        strcpy(description, "intern");
                }
                ~jobDescription(){
                        delete description;
        }
        char getDescription();

        void setDescription(char *jd);

};

char jobDescription::getDescription(){
    cout<<"description: ";
    cin>>description;
}

void jobDescription::setDescription(char *jd){
    description = new char[strlen(jd)+1];
    strcpy(description, jd);
}


//--------EMPLOYEE--------------------------//

class employee :  public division, public jobDescription, public spouse, public child {
    public:
        char *companyId;
        char *title;
        int startDate;


        employee(){
            companyId = new char;
            strcpy(companyId, "c1");
            title = new char;
            strcpy(title, "worker");
            startDate = 2018;
        }
        ~employee(){

            delete companyId;
            delete title;

        }
        char getCompanyId(){
            cout<<"company Id: ";
            cin>>companyId;
        }
        char getTitle(){
            cout<<"title: ";
            cin>>title;
        }
        int getStartDate(){
            cout<<"start Date: ";
            cin>>startDate;
        }

        char setCompanyId(char * cid){
            companyId = new char[strlen(cid)+1];
            strcpy(companyId, cid);
        }
        void setTitle(char * t){
            title = new char[strlen(t)+1];
            strcpy(title, t);
        }
        void setStartDate(int d){
            startDate = d;
        }



        void getEmployeeInfo(){
            cout<< "\nEmployeegiin basic infog oruulna uu: \n";
            getName();
            getSSNum();
            getAge();

            cout<< "\nEmployeegiin companygiin infog oruulna uu: \n";
            getCompanyId();
            getTitle();
            getStartDate();

            getDivisionName();
            getDescription();


        }

        void printEmployeeInfo(void){
            cout<<"\nemployeegin info: "<<endl;
            cout<<"Name: "<<name<<endl;
            cout<<"SSNum: "<<SSNum<<endl;
            cout<<"age: "<<age<<endl;

            cout<<"\nemployeegin companygiin info: "<<endl;
```

```
            cout<<"company Id: "<<companyId<<endl;
            cout<<"title: "<<title<<endl;
            cout<<"startDate: "<<startDate<<endl;
            cout<<"job Description: "<<description<<endl;


        }
};



//---------MAIN--------//

int main(){
    int n, d;
    int i;
    employee p;
    child ch1;
    child ch2;
    spouse s1;

    vector<child> NChild;

    p.getEmployeeInfo();
    cout<<"emoployee huuhedtei yu (1-yes/0-no)?";
    cin>>d;

    if (d==1){
        NChild.push_back( ch1 );
        NChild.push_back( ch2 );
        NChild[0].getfavoriteToy();


    }else{
        cout<<"employee huuhedgui";
    }

    cout<<"emoployee spousetai yu (1-yes/0-no)?";
    cin>>d;
    if (d==1){
        s1.getAnniversaryDate();
    }else{
        cout<<"employee spousegui\n";
    }
```

```
        p.printEmployeeInfo();
        cout<< "\nEmployeegiin ger buliin info: ";
        s1.printSpouseInfo();


        NChild[0].printChildInfo();
        NChild[1].printChildInfo();

}
```