

GÀ HẦM TÁO ĐỎ

Gà hầm táo đỏ vừa ngọt nước vừa ngọt thịt mà lại bổ dưỡng, món ăn này rất thích hợp cho mọi người, đặc biệt là người suy dinh dưỡng và trẻ nhỏ.

Nguyên liệu

- Đùi gà
 - Táo đỏ khô
 - Cà rốt
 - Nấm đông cô
 - Hạt sen
- (- Hạt nêm, tiêu, muối, đường, xì dầu, nước dùng gà)

Khối lượng nguyên liệu sử dụng phụ thuộc vào sự kết hợp của đầu bếp.

Chế biến:

Bước 1: Đùi gà khi mua về, rửa sạch với nước. Táo đỏ khô ngâm trong nước âm ẩm khoảng 10 phút. Nấm cắt bỏ phần gốc cứng và dơ, rửa thật sạch với nước, cà rốt gọt sạch bở vỏ, cắt từng miếng vừa ăn.

Bước 2: Chuẩn bị một cái nồi có kích thước vừa đủ nấu, cho hết đùi gà vào nồi và đỗ thêm nước lọc sao cho nước ngập hết cả mặt đùi gà trong nồi, Bắc nồi nấu đùi gà lên bếp bật lửa đun.

Bước 3: Khi nồi nước nấu gà sôi, bật lửa nhỏ, vớt phần bột lợn cợn trên mặt nước dùng ra cho nước dùng trong hơn và duy trì lửa hầm trong vòng khoảng 1 tiếng đồng hồ.

Bước 4: Cho thêm phần cà rốt, hạt sen và nấm vào nồi hầm, hầm nguyên liệu khoảng vài phút (15-20 phút) cho đến khi nào cà rốt chín mềm. Nêm nếm lại muối, hạt nêm và bột tiêu vào trong nồi, thấy nước đã có hương vị vừa hợp khẩu vị rồi thì tắt bếp.

Cho bảng Thành phần dinh dưỡng các loại thực phẩm - calo, protein, carb, fat như bên dưới:

| Thực phẩm (100g) | Carb (g) | Fat (g) | Fat bão hòa | Protein (g) | Calo (Kcal) |
|------------------|----------|---------|-------------|-------------|-------------|
| Gà | 0 | 15 | 4 | 17 | 211 |
| Hạt sen | 17 | 0 | 1 | 4 | 89 |
| Cà rốt | 10 | 3 | 0 | 1 | 41 |
| Táo đỏ | 3 | 1 | 0 | 1 | 52 |
| Nấm đông cô | 1 | 0 | 0 | 15 | 10 |

Giả sử thành phần dinh dưỡng của thực phẩm tươi và khô, sống và chín như nhau, bỏ

qua thành phần dinh dưỡng từ gia vị, nước dùng. Áp dụng kiến thức lập trình hướng

đổi tượng (kế thừa, đa hình) thiết kế sơ đồ chi tiết các lớp đổi tượng và xây dựng chương trình thực hiện các yêu cầu sau:

1. Nhập vai vào phụ bếp và chuẩn bị các nguyên liệu cần thiết (nhập vào từ bàn phím khối lượng khác 0 từng loại thực phẩm) để tạo nên món gà hầm táo đỏ này.
2. Nhập vào chỉ số qui định carb, fat, fat bão hòa, protein, calo cung cấp tối đa cho cơ thể trong 1 bữa ăn. Giả sử khi nấu xong bạn sẽ ăn hết và hấp thu 80% chất dinh dưỡng trong phần gà hầm đã nấu. Kiểm tra xem gà hầm trên đã đáp ứng đủ nhu cầu dinh dưỡng của bạn chưa? Nếu chưa cần bổ sung thêm chất gì? Bao nhiêu?
3. Theo Tổ chức Nông Lương Liên Hợp Quốc (FAO), nhu cầu calo tối thiểu của một người trung bình là xấp xỉ 1800 calo/ngày. Nếu trung bình mỗi người một ngày ăn 3 lần thì mỗi lần cần nạp 600 calo. Nếu ăn dư thì toàn bộ calo thừa sẽ bị tích thành mỡ, có nguy cơ béo phì. Nếu tránh béo phì thì bạn cần tập thể dục, đơn giản nhất đó là chạy bộ. Nếu chạy ở tốc độ thường, ở địa hình bằng phẳng thì trung bình 30 phút chạy sẽ đốt khoảng 145-150 calo. Hỏi với bữa ăn trên thì bạn cần chạy bộ trong bao lâu để tiêu thụ hết lượng calo dư thừa? Nếu không cần chạy bộ thì xuất 0.



Source code: <https://www.youtube.com/watch?v=TKidGhOB0ws&t=51s>

The screenshot shows a Java development environment with the following details:

- Project:** buoi1.1
- File Structure:** The project contains a src folder with several packages:
 - baTap_docGia
 - gaHamTaoDo
 - oop
 - QuanLiTrungTamTheHinh
 - Array.java
 - BieuThucLamDa_Stream_ArrayList.java
 - HinhChuNhac
 - Main.java
 - Main1.java
 - Main3.java
 - Main4.java
 - Numerology.java
 - Point
 - ZodiacSign.java
- Code View:** The main focus is on the NguyenLieu.java file, which is part of the gaHamTaoDo package. The code defines a class NguyenLieu with methods for inputting carb values and setting/getting carb, fat, and fatBm values.

```
1 package gaHamTaoDo;
2
3 public class NguyenLieu { 12 usages 5 inheritors new *
4     protected int carb; 7 usages
5     protected int fat; 7 usages
6     protected int fatBH; 7 usages
7     protected int protein; 7 usages
8     protected int calo; 7 usages
9     protected int khoiLuong; 3 usages
10    public NguyenLieu() { 5 usages new *
11    }
12
13    public void nhapO() { 5 usages new *
14
15        java.util.Scanner scanner = new java.util.Scanner(System.in);
16        System.out.println("Mời bạn nhập khối lượng: ");
17        this.KhoiLuong = scanner.nextInt();
18    }
19
20    public int getCarb() { 1 usage new *
21        return carb;
22    }
23
24    public void setCarb(int carb) { no usages new *
25        this.carb = carb;
26    }
27
28    public int getFat() { 1 usage new *
29        return fat;
30    }
31
32    public void setFat(int fat) { no usages new *
33        this.fat = fat;
34    }
35
36    public int getFatBH() { 1 usage new *
37        return fatBH;
38    }
39
40    public void setFatBH(int fatBH) { no usages new *
41        this.fatBH = fatBH;
42    }
43}
```

Project

- buoi1.1 D:\tan\java\buoi1.1
 - .idea
 - out
 - src
 - baiTap_docGia
 - DanhSachDocGia
 - DocGia
 - DocGia_NguoiLon
 - DocGia_TreEm
 - gaHamTaoDo
 - CaRot
 - Ga
 - HatSen
 - MonAn
 - NamDongCo
 - NguyenLieu
 - TaoDo
 - oop
 - QuanLiTrungTamTheHinh
 - Array1.java
 - BieuThucLamDa_Stream_ArrayList.java
 - HinhChuNhat
 - Main.java
 - Main1.java
 - Main3.java
 - Main4.java
 - Numerology.java
 - Point
 - ZodiacSign.java
- .gitignore

NguyenLieu.java Ga.java HatSen.java CaRot.java TaoDo.java NamDongCo.java

```

1 package gaHamTaoDo;
2
3 public class HatSen extends NguyenLieu{ 1 usage new *
4     public HatSen(){ 1 usage new *
5         this.carb=17;
6         this.fat=0;
7         this.fatBh=1;
8         this.protein=4;
9         this.calor=89;
10    }
11 }
```

Project

- buoi1.1 D:\tan\java\buoi1.1
 - .idea
 - out
 - src
 - baiTap_docGia
 - DanhSachDocGia
 - DocGia
 - DocGia_NguoiLon
 - DocGia_TreEm
 - gaHamTaoDo
 - CaRot
 - Ga
 - HatSen
 - MonAn
 - NamDongCo
 - NguyenLieu
 - TaoDo
 - oop
 - QuanLiTrungTamTheHinh
 - Array1.java
 - BieuThucLamDa_Stream_ArrayList.java
 - HinhChuNhat
 - Main.java
 - Main1.java
 - Main3.java
 - Main4.java
 - Numerology.java
 - Point
 - ZodiacSign.java
 - .gitignore
 - buoi1.1.iml
 - External Libraries
 - Scratches and Consoles

NguyenLieu.java Ga.java HatSen.java CaRot.java TaoDo.java NamDongCo.java

```

1 package gaHamTaoDo;
2
3 import java.util.ArrayList;
4 import java.util.List;
5
6 public class MonAn { new *
7     private List<NguyenLieu> arr; 7 usages
8     public MonAn(){ 1 usage new *
9         this.arr=new ArrayList<>();
10    }
11    public void nhap(){ 1 usage new *
12        System.out.println("Mời bạn nhập gà: ");
13        NguyenLieu ga = new Ga();
14        ga.nhap();
15        this.arr.add(ga);
16
17        System.out.println("Mời bạn nhập hạt sen: ");
18        NguyenLieu hatsen = new HatSen();
19        hatsen.nhap();
20        this.arr.add(hatsen);
21
22        System.out.println("Mời bạn nhập cà rốt: ");
23        NguyenLieu carot = new CaRot();
24        carot.nhap();
25        this.arr.add(carot);
26
27        System.out.println("Mời bạn nhập táo đỏ: ");
28        NguyenLieu taodo = new TaoDo();
29        taodo.nhap();
30        this.arr.add(taodo);
31
32        System.out.println("Mời bạn nhập nấm đông cô: ");
33        NguyenLieu nam = new NamDongCo();
34        nam.nhap();
35        this.arr.add(nam);
36    }
37    public float tinhNangLuong(int flag){ 5 usages new *
38        float sum=0;
39        for(NguyenLieu a:arr)
40            if(flag==1)
41                sum+=a.getCarbohydrates();
42            else if(flag==2)
43                sum+=a.getFats();
44            else if(flag==3)
45                sum+=a.getProteins();
46            else if(flag==4)
47                sum+=a.getCalories();
48        return sum;
49    }
50 }
```

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project Tree:** The project is named "buoi1.1". It contains a ".idea" folder, an "out" folder, and a "src" folder. The "src" folder contains several Java files: "baiTap_docGia", "gaHamTaoDo", "MonAn", "NamDongCo", "NguyenLieu", and "TaoDo".
- Editor:** The "MonAn.java" file is currently selected and displayed in the editor. The code implements a class "MonAn" with methods for input ("nhapO") and calculation ("tinhNangLuong").
- Run Configuration:** A run configuration for "gaHamTaoDo.MonAn" is selected in the bottom-left corner.

```
public class MonAn { new *
    public void nhapO(){ 1 usage new *
        }
        public float tinhNangLuong(int flag){ 5 usages new *
            float sum=0;
            for(NguyenLieu a:this.arr){
                if(flag==1){
                    sum+=a.getKhoiLuong()/100.0f*a.getCarb();
                }
                else if(flag==2){
                    sum+=a.getKhoiLuong()/100.0f*a.getFat();
                }
                else if(flag==3){
                    sum+=a.getKhoiLuong()/100.0f*a.getFatBH();
                }
                else if(flag==4){
                    sum+=a.getKhoiLuong()/100.0f*a.getProtein();
                }
                else if(flag==5){
                    sum+=a.getKhoiLuong()/100.0f*a.getCalo();
                }
            }
            return sum*0.8f;
        }
        public void cau2(){ 1 usage new *
            java.util.Scanner scanner = new java.util.Scanner(System.in);
            System.out.println("Mời bạn nhập chỉ số carb cần thiết cho 1 bữa ăn ");
            int carb= scanner.nextInt();
            float nangLuong=tinhNangLuong( flag:1 );
            if(carb> nangLuong){
                System.out.printf("Có thể thêm %f carb nữa nha",carb -nangLuong);
            }
            System.out.println("Mời bạn nhập chỉ số fat cần thiết cho 1 bữa ăn ");
            int fat= scanner.nextInt();
            nangLuong=tinhNangLuong( flag:2 );
            if(fat> nangLuong){
                System.out.printf("Có thể thêm %f fat nữa nha",fat -nangLuong);
            }
        }
    }
```

```
37     public float tinhNangLuong(int flag){ 5 usages new *
38         return sum*0.8f;
39     }
40     public void cau2(){ 1 usage new *
41         java.util.Scanner scanner = new java.util.Scanner(System.in);
42         System.out.println("Mời bạn nhập chỉ số carb cần thiết cho 1 bữa ăn ");
43         int carb= scanner.nextInt();
44         float nangLuong=tinhNangLuong( flag: 1);
45         if(carb> nangLuong){
46             System.out.printf("Cơ thể em cần bổ sung thêm %f carb nữa nha",carb -nangLuong);
47         }
48
49         System.out.println("Mời bạn nhập chỉ số fat cần thiết cho 1 bữa ăn ");
50         int fat= scanner.nextInt();
51         nangLuong=tinhNangLuong( flag: 2);
52         if(fat> nangLuong){...}
53
54         System.out.println("Mời bạn nhập chỉ số fat bh cần thiết cho 1 bữa ăn ");
55         int fatbh= scanner.nextInt();
56         nangLuong=tinhNangLuong( flag: 3);
57         if(fatbh> nangLuong){...}
58
59         System.out.println("Mời bạn nhập chỉ số protein bh cần thiết cho 1 bữa ăn ");
60         int protein= scanner.nextInt();
61         nangLuong=tinhNangLuong( flag: 4);
62         if(protein> nangLuong){
63             System.out.printf("Cơ thể em cần bổ sung thêm %f protein nữa nha",protein -nangLuong);
64         }
65
66         System.out.println("Mời bạn nhập chỉ số calo bh cần thiết cho 1 bữa ăn ");
67         int calo= scanner.nextInt();
68         nangLuong=tinhNangLuong( flag: 5);
69         if(calo> nangLuong){
70             System.out.printf("Cơ thể em cần bổ sung thêm %f calo nữa nha",calo -nangLuong);
71         }
72     }
73
74     public static void main(String[] args) { new *
75         MonAn a=new MonAn();
76         a. nhap();
```

Đầu những năm 1900, dựa trên sự hiện diện của các kháng nguyên trên màng hồng cầu, các nhà khoa học đã xác định rằng con người có 4 nhóm máu khác nhau: O, A, B và AB. Hệ thống phân loại nhóm máu này (gọi là hệ thống nhóm máu ABO) cung cấp cho bác sĩ các thông tin quan trọng để lựa chọn nhóm máu phù hợp trong việc truyền máu. Và đồng thời có thể tiên đoán được nhóm máu tương đối của người con dựa trên nhóm máu của cha mẹ theo cơ chế di truyền học.

Nhóm máu của người con khi biết được nhóm máu của cha và mẹ

| | | Nhóm máu người cha | | | | Dự đoán khả năng nhóm máu người con |
|-------------------|----|--------------------|-----------------|--------------|----------|-------------------------------------|
| Nhóm máu người mẹ | A | B | AB | O | | |
| | A | A hoặc O | A, B, AB hoặc O | A, B hoặc AB | A hoặc O | |
| | B | A, B, AB hoặc O | B hoặc O | A, B hoặc AB | B hoặc O | |
| | AB | A, B hoặc AB | A, B hoặc AB | A, B hoặc AB | A hoặc B | |
| | O | A hoặc O | B hoặc O | A hoặc B | O | |

Ngoài ra còn có thêm hệ thống phân loại Rh (Rhesus)

Căn cứ vào sự khác biệt khi nghiên cứu về sự vận chuyển oxy của hồng cầu thì các hồng cầu có thể mang ở mặt ngoài một protein gọi là Rhesus. Nếu có kháng nguyên D thì là nhóm Rh⁺ (dương tính), nếu không có là Rh⁻ (âm tính). Các nhóm máu A, B, O, AB mà Rh⁻ thì được gọi là âm tính A⁻, B⁻, O⁻, AB⁻. Nhóm máu Rh⁻ chỉ chiếm 0,04% dân số thế giới. Đặc điểm của nhóm máu Rh này là chúng chỉ có thể nhận và cho người cùng nhóm máu, đặc biệt phụ nữ có nhóm máu Rh⁻ thì con rất dễ tử vong.

Người có nhóm máu Rh⁺ chỉ có thể cho người cũng có nhóm máu Rh⁺ và nhận người có nhóm máu Rh⁺ hoặc Rh⁻.

Người có nhóm máu Rh⁻ có thể cho người có nhóm máu Rh⁺ hoặc Rh⁻ nhưng chỉ nhận được người có nhóm máu Rh⁻ mà thôi
Trường hợp người có nhóm máu Rh⁻ được truyền máu Rh⁺, trong lần đầu tiên sẽ không có bất kỳ phản ứng tức thì nào xảy ra nhưng nếu tiếp tục truyền máu Rh⁺ lần thứ 2 sẽ gây ra những

hậu quả nghiêm trọng do tai biến truyền máu. Tương tự với trường hợp mẹ Rh- sinh con (lần đầu và lần thứ hai trờ đi)

Khả năng tương thích: ✓: Có thể cho - nhận.

✗: Không thể cho - nhận.

| Người nhận | Bảng khả năng tương thích hồng cầu | | | | | | | |
|------------|------------------------------------|----|----|----|----|----|-----|-----|
| | O- | O+ | A- | A+ | B- | B+ | AB- | AB+ |
| O- | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| O+ | ✓ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| A- | ✓ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ |
| A+ | ✓ | ✓ | ✓ | ✓ | ✗ | ✗ | ✗ | ✗ |
| B- | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ |
| B+ | ✓ | ✓ | ✗ | ✗ | ✓ | ✓ | ✗ | ✗ |
| AB- | ✓ | ✗ | ✓ | ✗ | ✓ | ✗ | ✓ | ✗ |
| AB+ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |

Áp dụng kiến thức lập trình hướng đối tượng (kế thừa, đa hình) thiết kế sơ đồ chi tiết các lớp đối tượng (1.5đ) và xây dựng chương trình thực hiện các yêu cầu sau:

- Nhập danh sách các nhóm máu của một nhóm người. (1đ)
- Cho một bộ 3 nhóm máu của 3 người là cha, mẹ, con. Hãy kiểm tra và đưa ra kết quả nhóm máu có phù hợp với quy luật di truyền hay không? (1đ)
- Chọn một người X trong danh sách. Hãy liệt kê tất cả các người còn lại trong danh sách có thể cho máu người X này. (1đ)

Lưu ý: Trong trường hợp sinh viên không biết về nhóm máu và di truyền học trước đây thì phải đọc kỹ thông tin trên (các thông tin trên đủ để sinh viên thực hiện các yêu cầu của đề thi) và nghiêm túc làm bài. Giám thị coi thi không giải thích gì thêm.

-----HẾT-----



```
public float tinhNangLuong2(int flag){ no usages new *
    return (float) this.arr.stream() Stream<NguyenLieu>
        .mapToDouble(a ->{
            if(flag==FLAG_CARB){
                return a.getKhoiLuong()/100.0f*a.getCarb();
            }
            else if(flag==FLAG_FAT){
                return a.getKhoiLuong()/100.0f*a.getFat();
            }
            else if(flag==FLAG_FAT_BH){
                return a.getKhoiLuong()/100.0f*a.getFatBH();
            }
            else if(flag==FLAG_PROTEIN){
                return a.getKhoiLuong()/100.0f*a.getProtein();
            }
            else if(flag==FLAG_CALO){
                return a.getKhoiLuong()/100.0f*a.getCalo();
            }
            throw new IllegalArgumentException("Invalid flag value");
        }) DoubleStream
        .sum();
}
```

```
}

public float cau3(){ 1usage new *
    if(tinhNangLuong(FLAG_CAL0)-600>0){
        return 30*(tinhNangLuong(FLAG_CAL0)-600)/147.5f;
    }
    else{
        return 0;
    }
}

public static void main(String[] args) { new *
    MonAn a=new MonAn();
    a. nhap();
    a.cau2();
    System.out.println("câu 3: "+a.cau3());
}
```

```
}
```

```
NguyenLieu.java  Ga.java  HatSen.java  Carot.java  TaoDo.java  NamDongCo.java
package gaHamTaoDo;

import java.util.ArrayList;
import java.util.List;

▷ public class MonAn { new *
    public static final int FLAG_CAR = 1; 2 usages
    public static final int FLAG_FA = 2; 2 usages
    public static final int FLAG_FA_BH = 3; 2 usages
    public static final int FLAG_PRTEIN = 4; 2 usages
    public static final int FLAG_CALO = 5; 4 usages
    private List<NguyenLieu> arr; 8 usages
    public MonAn(){ 1 usage new *
        this.arr=new ArrayList<>();
    }
    public void nhap(){ 1 usage new *
        System.out.println("Mời bạn nhập gà: ");
        NguyenLieu ga = new Ga();
        ga.nhap();
        this.arr.add(ga);
    }
}
```

[Danh sách điểm kt TOPCIT.pdf - Google Drive](#)

Danh sách điểm kt TOPCIT:

https://docs.google.com/spreadsheets/d/1rb1rUh2QIQ7alv_IrSifRUcaeEiocBg6-y5VN5BTt3E/edit?usp=sharing

The screenshot shows a Java development environment with the following details:

- Project Structure:** The project is named "buoi1.1". It contains a "src" directory with several packages:
 - Baitap_NhomMau:** Contains classes A, AB, B, DanhSachNhomMau, NhomMau, and O.
 - ganhantodo:** Contains classes CaRot, Ga, HatSen, MonAn, NamDongCo, NguyenLieu, and TaoDo.
 - oop:** Contains classes Main, Main1, Main3, Main4, Numerology, Point, and ZodiacSign.
 - QuanLiTrungTamTheHinh:** Contains classes Array, BleuThucLamDa_Stream_ArrayList, and HinhChuHinh.
- Code Editor:** The right pane displays the code for the **NhomMau.java** file. The code defines a class **NhomMau** with protected fields **loaiMau** (int) and **rh** (boolean). It includes methods **nhap()**, **xuat()**, and **setRh(boolean rh)**. The **xuat()** method prints the value of **rh** based on the value of **loaiMau**. The **setRh(boolean rh)** method sets the value of **rh**.
- Run Tab:** The "Run" tab is active, showing the output of the application: "true".

The screenshot shows a Java development environment with a dark theme. On the left is the Project Explorer, displaying a file tree for a project named 'buoi1.1'. The 'src' folder contains several packages: 'balTap_docGia' (with classes 'DanhSachDocGia', 'DocGia', 'DocGia_NguoiLon', 'DocGia_TreEm'), 'BaitapNhomMau' (with classes 'A', 'AB', 'B', 'DanhSachNhomMau', 'NhomMau', 'O'), 'gaHamTaoDo' (with classes 'CaRot', 'Ga', 'HatSen', 'MonAn', 'NamDongCo', 'NguyenLieu', 'TaoDo'), and others like 'oop', 'QuanLiTrungTamTheHinh', 'Main.java', 'Main1.java', 'Main3.java', 'Main4.java'. The 'out' folder is also visible. On the right is the code editor, showing the content of the 'A' class from the 'BaitapNhomMau' package. The code is as follows:

```
1 package BaitapNhomMau;
2
3 public class A extends NhomMau { no usages new *
4     public A(){ no usages new *
5         this.loaiMau=_A;
6     }
7 }
8
```

File structure:

```

buoi1.1 D:\tan\java\buoi1.1
  +-- .idea
  +-- out
  +-- src
    +-- baiTap_docGia
      +-- DanhSachDocGia
      +-- DocGia
      +-- DocGia_NguoiLon
      +-- DocGia_TreEm
    +-- BaiTapNhomMau
      +-- A
      +-- AB
      +-- B
      +-- DanhSachNhomMau
      +-- NhomMau
      +-- O
    +-- gaHamTaoDo
      +-- CaRot
      +-- Ga
      +-- HatSen
      +-- MonAn
      +-- NamDongCo
      +-- NguyenLieu
      +-- TaoDo
    +-- oop
    +-- QuanLiTrungTamTheHinh
      +-- Array.java
      +-- BieuThuCuaListDa_Stream_ArrayList.java
      +-- HinChuNhat
      +-- Main.java
      +-- Main1.java
      +-- Main3.java
      +-- Main4.java
      +-- Numerology.java
      +-- Point
      +-- ZodiacSign.java
    .gitignore

```

Code (DanhSachNhomMau.java):

```

import java.util.List;

public class DanhSachNhomMau {
    private List<NhomMau> arr;
    public DanhSachNhomMau() {
        this.arr = new ArrayList<>();
    }
    public void nhap() {
        java.util.Scanner scanner = new java.util.Scanner(System.in);
        System.out.println("Mỗi bạn nhập số lượng: ");
        int n = scanner.nextInt();
        for(int i=0;i<n;i++){
            System.out.println("Nhập thông tin người thứ "+i+1);
            System.out.println("1. Nhóm 0");
            System.out.println("2. Nhóm A");
            System.out.println("3. Nhóm B");
            System.out.println("4. Nhóm AB");
            System.out.println("Mỗi bạn lựa chọn nhóm máu");
            int flag = scanner.nextInt();
            if(flag==1){
                NhomMau x=new O();
                x.nhap();
                this.arr.add(x);
            } else if(flag==2){
                NhomMau x=new A();
                x.nhap();
                this.arr.add(x);
            } else if(flag==3){
                NhomMau x=new B();
                x.nhap();
                this.arr.add(x);
            } else if(flag==4){
                NhomMau x=new AB();
                x.nhap();
                this.arr.add(x);
            }
        }
    }
}

```

File structure:

```

Project ▾
  buoi1.1 D:\tan\java\buoi1.1
    +-- .idea
    +-- out
    +-- src
      +-- baiTap_docGia
        +-- DanhSachDocGia
        +-- DocGia
        +-- DocGia_NguoiLon
        +-- DocGia_TreEm
      +-- BaiTapNhomMau
        +-- A
        +-- AB
        +-- B
        +-- DanhSachNhomMau
        +-- NhomMau
        +-- O
      +-- gaHamTaoDo
        +-- CaRot
        +-- Ga
        +-- HatSen
        +-- MonAn
        +-- NamDongCo
        +-- NguyenLieu
        +-- TaoDo
      +-- oop
      +-- QuanLiTrungTamTheHinh
        +-- Array.java
        +-- BieuThuCuaListDa_Stream_ArrayList.java
        +-- HinChuNhat
        +-- Main.java
        +-- Main1.java
        +-- Main3.java
        +-- Main4.java
        +-- Numerology.java
        +-- Point

```

Code (DanhSachNhomMau.java):

```

public class DanhSachNhomMau {
    private List<NhomMau> arr;
    public void nhap() {
        x.nhap();
        this.arr.add(x);
    }
    public void xuat(){
        for(NhomMau x:this.arr){
            x.xuat();
        }
    }
    public static void main(String[] args) {
        DanhSachNhomMau a=new DanhSachNhomMau();
        a.nhap();
        a.xuat();
    }
}

```

Kiểm tra di truyền và cho nhận

```
public abstract class NhomMau { 32 usages 4 inheritors new *
    public void xuat(){ new *
        System.out.print("+");
    }
    System.out.println("#####");
}
public abstract boolean KiemTraDiTruyen(NhomMau cha, NhomMau con); 1 usage 4 implementations new *
public abstract boolean KiemTraNhomMau(NhomMau nhan); 1 usage new *
public boolean KiemTraRh(NhomMau nhan){ 1 usage new *
    if(this.rh == true){
        if (nhan.rh == true){
            return true;
        }
    } else if (this.rh == false) {
        return true;
    }
    return false;
}

public boolean KiemTraChoNhan(NhomMau nhan){ 1 usage new *
    boolean nhomMau = KiemTraNhomMau(nhan);
    boolean rh = this.KiemTraRh(nhan);
    if (nhomMau && rh){
        return true;
    } else {
        return false;
    }
}

public boolean isRh() { no usages new *
    return rh;
}

public void setRh(boolean rh) { no usages new *
```

```
package BaitapNhomMau;

public class A extends NhomMau { 5 usages new *
    public A(){ 5 usages new *
        this.loaiMau=_A;
    }
    @Override 1 usage new *
    public boolean KiemTraDiTruyen(NhomMau cha, NhomMau con){
        if(cha.getLoaiMau() == _O || cha.getLoaiMau() == _A){
            if(con.getLoaiMau() == _A || con.getLoaiMau() == _O){
                return true;
            }
        } else if(cha.getLoaiMau() == _B){
            return true;
        } else if(cha.getLoaiMau() == _AB){
            if(con.getLoaiMau() != _O){
                return true;
            }
        }
        return false;
    }

    public boolean KiemTraNhomMau(NhomMau nhan){ 1 usage new *
        if (nhan.getLoaiMau() == _A || nhan.getLoaiMau() == _AB){
            return true;
        }
    }
}
```

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project Tree:** The project is named "buoi1.1". It contains several packages:
 - src**:
 - baiTap_docGia**:
 - DanhSachDocGia
 - DocGia
 - DocGia_NguoiLon
 - DocGia_TreEm
 - BaitapNhomMau**:
 - A
 - AB
 - B**
 - DanhSachNhomMau
 - NhomMau
 - O
 - gaHamTaoDo**:
 - CaRot
 - Ga
 - HatSen
 - MonAn
 - NamDongCo
 - NguyenLieu
 - TaoDo
 - oop**
 - QuanLiTrungTamTheHinh**:
 - ArrayList.java
 - BieuThucLamDa_Stream_ArrayList.java
 - HinhChuNhat**
 - Main.java**
 - Main1.java**
 - Main3.java**
 - Main4.java**
 - Code Editor:** The current file is "B.java" under the "BaitapNhomMau" package. The code implements the `NhomMau` interface and checks if two objects have the same category (A, AB, or B). It also includes a method to check if two `NhomMau` objects are equal.

```
package BaitapNhomMau;

public class B extends NhomMau { 2 usages new *
    public B(){ 2 usages new *
        this.loaiMau=_B;
    }
    public boolean KiemTraDiTruyen(NhomMau cha, NhomMau con) { 1 usage new *
        if(cha.getLoaiMau()==_O || cha.getLoaiMau()==_B ){
            if(con.getLoaiMau()==_B || con.getLoaiMau()==_O){
                return true;
            }
        }
        else if(cha.getLoaiMau()==_A){
            return true;
        }
        else if(cha.getLoaiMau()==_AB){
            if(con.getLoaiMau()!=_O){
                return true;
            }
        }
        return false;
    }

    public boolean KiemtraNhomMau(NhomMau nhan){ 1 usage new *
        if (nhan.getLoaiMau() == _B || nhan.getLoaiMau() == _AB){
            return true;
        } return false;
    }
}
```

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project Tree:** The project is named "buoi1.1". It contains several packages:
 - src**:
 - baiTap_docGia**:
 - DanhSachDocGia
 - DocGia
 - DocGia_NguoiLon
 - DocGia_TreEm
 - BaitapNhomMau**:
 - A
 - AB**
 - B
 - DanhSachNhomMau
 - NhomMau
 - O
 - gaHamTaoDo**:
 - CaRot
 - Ga
 - HatSen
 - MonAn
 - NamDongCo
 - NguyenLieu
 - TaoDo
 - oop**
 - QuanLiTrungTamTheHinh**:
 - ArrayList.java
 - BieuThucLamDa_Stream_ArrayList.java
 - HinhChuNhat**
 - Main.java**
 - Main1.java**
 - Main3.java**
 - Main4.java**
 - Code Editor:** The current file is "AB.java" under the "BaitapNhomMau" package. The code implements the `NhomMau` interface and checks if two objects have the same category (A, AB, or B). It also includes a method to check if two `NhomMau` objects are equal.

```
package BaitapNhomMau;

public class AB extends NhomMau { 1 usage new *
    public AB(){ 1 usage new *
        this.loaiMau=_AB;
    }
    public boolean KiemTraDiTruyen(NhomMau cha, NhomMau con) { 1 usage new *
        if(cha.getLoaiMau() !=_O){
            if(con.getLoaiMau()!=_O){
                return true;
            }
        }
        else if(cha.getLoaiMau()==_O){
            if(con.getLoaiMau()==_A|| con.getLoaiMau()==_B){
                return true;
            }
        }
        return false;
    }

    public boolean KiemtraNhomMau(NhomMau nhan){ 1 usage new *
        if (nhan.getLoaiMau() == _AB){
            return true;
        } return false;
    }
}
```

The screenshot shows a Java development environment with a file tree on the left and a code editor on the right.

File Tree:

- .idea
- out
- src
 - baiTap_docGia
 - DanhSachDocGia
 - DocGia
 - DocGia_NguoiLon
 - DocGia_TreEm
 - BaitapNhomMau
 - A
 - AB
 - B
 - DanhSachNhomMau
 - NhomMau
 - O
 - gaHamTaoDo
 - CaRot
 - Ga
 - HatSen
 - MonAn
 - NamDongCo
 - NguyenLieu
 - TaoDo
 - oop
 - QuanLiTrungTamTheHinh
 - ArrayList.java
 - BieuThuClamDa_Stream_ArrayList.java
 - HinhChuNhac
 - Main.java
 - Main1.java
 - Main3.java
 - Main4.java
 - Numerology.java
 - Point
 - TeddyBear.java

Code Editor (Class O.java):

```
2
3 public class O extends NhomMau { 1 usage new *
4     public O() { 1 usage new *
5         this.loaiMau = _O;
6     }
7
8     public boolean KiemTraDiTruyen(NhomMau cha, NhomMau con) { 1 usage new *
9         if (cha.getLoaiMau() == _A) {
10             if (con.getLoaiMau() == _A || con.getLoaiMau() == _O) {
11                 return true;
12             }
13         } else if (cha.getLoaiMau() == _B) {
14             if (con.getLoaiMau() == _B || con.getLoaiMau() == _O) {
15                 return true;
16             }
17         } else if (cha.getLoaiMau() == _AB) {
18             if (con.getLoaiMau() == _A || con.getLoaiMau() == _B) {
19                 return true;
20             }
21         }
22     } else if (cha.getLoaiMau() == _O) {
23         if (con.getLoaiMau() == _O) {
24             return true;
25         }
26     }
27     return false;
28 }
29
30
31
32     public boolean KiemtraNhomMau(NhomMau nhan){ 1 usage new *
33         return true;
34     }
35
36 }
37
```

```
⑤ NguyenLieu.java    ⑤ HatSen.java    ⑤ MonAn.java    ⑤ NhomMau.java    ⑤ O.java    ⑤ A.java    ⑤ DanhSachNhomMau
6     public class DanhSachNhomMau { new *
46     public void xuat(){ new *
51     }
52     public void cau2(){ no usages new *
53         NhomMau me=new A();
54         NhomMau cha=new B();
55         NhomMau con=new A();
56         if(me.KiemTraDiTruyen(cha,con)){
57             System.out.println("Gia đình này phù hợp quy luật di truyền");
58         }
59         else{
60             System.out.println("Có gì đó sai sai");
61         }
62     }
63     public void cau3(){ 1 usage new *
64         java.util.Scanner scanner = new java.util.Scanner(System.in);
65         System.out.println("Mỗi bạn chọn 1 người để nhận máu");
66         int index = scanner.nextInt();
67         for(var i=0;i<arr.size();i++){
68             if(i !=index){
69                 if(this.arr.get(i).KiemtraChoNhan(this.arr.get(index))){
70                     System.out.printf("Bạn thứ %d có thể cho máu bạn thứ %d nhé \n",i,index);
71                 }
72             }
73         }
74     }
75 }
76 }
77 ▷ public static void main(String[] args){ new *
78     DanhSachNhomMau a=new DanhSachNhomMau();
79     a.nhap();
80     a.xuat();
81     // a.cau2();
82     a.cau3();
83 }
84 }
85 }
86 }
```