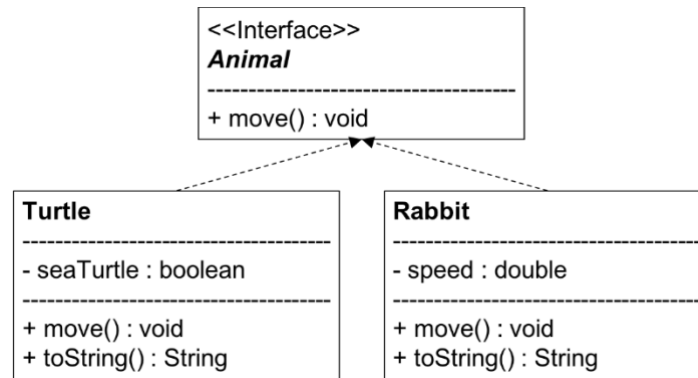


CẤU TRÚC DỮ LIỆU & GIẢI THUẬT 1

Ôn Tập Cuối Kỳ

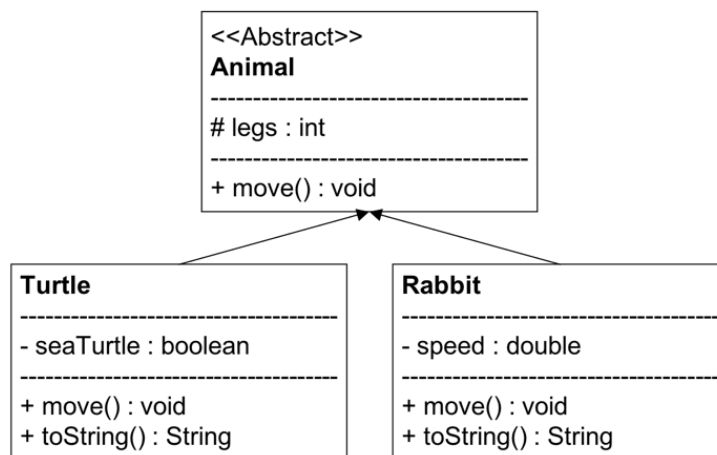
Câu 1. Dựa vào sơ đồ lớp như sau, anh (chị) hãy hiện thực chương trình bằng ngôn ngữ lập trình Java.



Yêu cầu:

- Tương ứng mỗi lớp dẫn xuất, anh (chị) hiện thực đầy đủ 3 phương thức khởi tạo, gồm: phương thức khởi tạo không tham số, phương thức khởi tạo có đầy đủ tham số, và phương thức khởi tạo sao chép.
- Hiện thực phương thức `move()` bằng cách in ra màn hình chuỗi đại diện loài động vật di chuyển, cụ thể:
 - Lớp Turtle: *"Turtle is running"*.
 - Lớp Rabbit: *"Rabbit is running"*.
- Hiện thực phương thức `toString()` trả về chuỗi chứa thông tin của lớp dẫn xuất tương ứng.

Câu 2. Dựa vào sơ đồ lớp như sau, anh (chị) hãy hiện thực chương trình bằng ngôn ngữ lập trình Java.



Yêu cầu:

- Tương ứng mỗi lớp dẫn xuất, anh (chị) hiện thực đầy đủ 3 phương thức khởi tạo, gồm: phương thức khởi tạo không tham số, phương thức khởi tạo có đầy đủ tham số, và phương thức khởi tạo sao chép.
- Hiện thực phương thức `move()` bằng cách in ra màn hình chuỗi đại diện loài động vật di chuyển, cụ thể:
 - Lớp Turtle: *"Turtle is running"*.
 - Lớp Rabbit: *"Rabbit is running"*.
- Hiện thực phương thức `toString()` trả về chuỗi chứa thông tin của lớp dẫn xuất tương ứng.

Câu 3. Áp dụng kỹ thuật lập trình đệ quy, anh (chị) hãy hiện thức phương thức tính giá trị các biểu thức sau:

$$a. A(n) = \begin{cases} 1, & n = 1 \\ 2A(n-1) + \frac{1}{2}, & n > 1 \end{cases}$$

$$b. B(n) = \begin{cases} \frac{2}{3}, & n = 0 \\ \sqrt{2} + B(n-1), & n > 0 \end{cases}$$

$$c. C(n) = \begin{cases} \frac{2}{3}, & n = 0 \\ \frac{3}{4}, & n = 1 \\ 2\sqrt{3}C(n-2) + C(n-1), & n > 1 \end{cases}$$

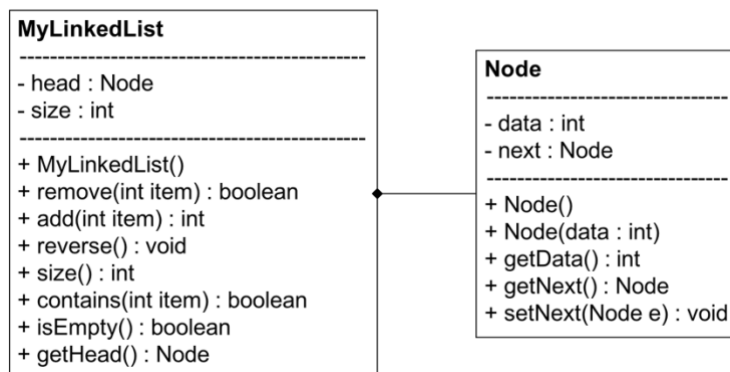
$$d. D(m, n) = \begin{cases} 2n + 3; & m = 0 \\ D(m-1, 0); & m > 0, n = 0 \\ D(2m-2, n-2); & m > 0, n > 0 \end{cases}$$

Câu 4. Cho dãy số sau:

17 15 3 23 4 1 32 9 18 25

Anh (chị) hãy trình bày từng bước quá trình sắp xếp dãy số sau bằng giải thuật *Insertion Sort*, *Bubble Sort*, *Selection Sort*, *Merge Sort*.

Câu 5. Một danh sách liên kết chứa các phần tử là số nguyên sắp xếp theo thứ tự tăng dần, có đặc tả như sau:



Yêu cầu:

- Anh (chị) hãy hiện thực phương thức *add(int item):int* để thêm một phần tử mới vào danh sách liên kết. Phương thức trả về số lượng phần tử hiện tại trong danh sách.

Ví dụ: cho danh sách $L = \{3, 6, 12, 14, 17, 23\}$

- Sau khi thêm 15; trả về 7 và danh sách $L = \{3, 6, 12, 14, 15, 17, 23\}$,
- Sau khi thêm 2, trả về 8 và danh sách $L = \{2, 3, 6, 12, 14, 15, 17, 23\}$,
- Sau khi thêm 34, trả về 9 và danh sách $L = \{2, 3, 6, 12, 14, 15, 17, 23, 34\}$,
- Sau khi thêm 2, trả về 9 và danh sách $L = \{2, 3, 6, 12, 14, 15, 17, 23, 34\}$.

- Anh (chị) hãy hiện thực phương thức *remove(int item):boolean* để xóa phần tử có giá trị *item* trong danh sách. Phương thức trả về *true* nếu xóa thành công, ngược lại, trả về *false*.

Ví dụ: cho danh sách $L = \{3, 6, 12, 14, 17, 23\}$

- Sau khi xóa 15; trả về *false* và danh sách $L = \{3, 6, 12, 14, 17, 23\}$,
- Sau khi xóa 12; trả về *true* và danh sách $L = \{3, 6, 14, 17, 23\}$.

- c. Anh (chị) hãy hiện thực phương thức *reverse():void* để đảo chiều danh sách liên kết. Lưu ý: không sử dụng danh sách tạm hay mảng tạm.

Ví dụ: cho danh sách $L = \{3, 6, 12, 14, 17, 23\}$; kết quả $L_{reversed} = \{23, 17, 14, 12, 6, 3\}$.

Câu 6. Anh (chị) hãy cho biết độ phức tạp (Big-O) của các chương trình sau:

- a.

```
for(i = 0 ; i < n; i++) {
    for(j = 0; j < n; j++) {
        for(k = n + 1; k >= 0; k--) {
            index = 1;
            while(index < n) {
                print("Hello");
                index *= 2;
            }
        }
    }
}
```
- b.

```
sum = 0, start = n;
while(start >= 0)
{
    for(int i = n*n - 1; i >= 0; i--) {
        sum += i*3;
    }
    start = start/5;
}
```
- c.

```
mul = 1;
for(int i = n - 1; i >= 0; i--)
{
    for(int j = 0; j < n ; j++)
    {
        k = n;
        while(k >= 0)
        {
            mul *= k;
            k /= 3;
        }
    }
}
```
- d.

```
k = 0;
for(i = 1; i <= n*n; i++)
{
    j = 1;
    while(j < n)
    {
        k++;
        j = j * 2;
    }
}
```
- e.

```
sum = 0;
for (i = n*n; i > 0; --i)
{
    for (j = n - 1; j >= 0; --j)
    {
```

```

        tmp = n - 1;
        while (tmp > 0)
        {
            sum = sum + (i/j);
            tmp /= 2;
        }
    }
}
f. int i,j;
   for(i = 0; i < n; ++i)
   {
       for(j = 1; j <= n - 1; ++j)
       {
           print("Good luck!");
       }
   }

```

Câu 7. Cho bảng băm có kích thước là 11. Anh (chị) hãy lần lượt thêm các giá trị sau vào bảng băm (với $h(g) = g \bmod 11$):

21, 8, 5, 55, 12, 7, 19, 67, 11, 33

Trong trường hợp xảy ra đụng độ, anh (chị) dùng kỹ thuật dò tuyến tính.

Câu 8. Cho bảng băm có kích thước là 7. Anh (chị) hãy lần lượt thêm các giá trị sau vào bảng băm (với $h(g) = g \bmod 7$):

14, 25, 5, 32, 26, 6

Trong trường hợp xảy ra đụng độ, anh (chị) dùng kỹ thuật dò tuyến tính.