

Báo cáo

Bài tập thực hành Trò chơi: đoán tên con vật

I.Cấu trúc dữ liệu:

Node: gồm 1 string chứa dữ liệu là 1 câu hỏi #Q hoặc 1 câu trả lời #A cho người chơi. Con trỏ left, right khởi tạo NULL để nối với tối đa 2 node khác và tạo cây nhị phân.

newQA: tạo thêm node mới

II.Hàm được sử dụng (có chú thích trong hàm):

1. QA* newQA(string data)

```
QA* newQA(string data)// tao node moi
{
    QA* tmp = new QA;
    tmp->sen = data;
    return tmp;
}
```

2. QA* input(QA* &root)

```
QA* input(QA* &root )// Phan 1: doc du lieu tu file thanh cay BST
{
    if (fs.eof()) return NULL;
    string tmp;
    getline(fs, tmp);
    root = newQA(tmp);
    if (tmp.substr(0, 2) == "#A") return root;// Neu la nodee la thi khong di tiep
    if (root->left==NULL) root->left = input(root->left);// tiep tuc di xuong cay con trai va phai
    if (root->right==NULL) root->right = input(root->right);
    return root;
}
```

3. void update(QA* &temp, vector<string> vans)

```
void update(QA* &temp, vector<string> vans)//Cap nhat them con vat moi tu nguoi choi
{
    cout << "I don't know it!\n";
    cout << "Would you like to teach me? (y/n): ";
    char choice;
    cin >> choice;
    if (choice == 'n') return;
    cout << "I asked the following: \n";
    for (int i = 0; i < vans.size(); i++) cout << vans[i];    //lap lai cau hoi
    string ques, ans;
    cout << "Enter a new animal in the form of a question, e.g., 'Is it a whale?': ";
    cin.ignore();
    getline(cin, ans);
    cout << "Now enter a question for which the answer is 'yes' for your new animal,\nand which does not contradict your previous answers: ";
    getline(cin, ques);
    temp->right = newQA(temp->sen);//chuyen cau tra loi qua node phai
    temp->sen = "#Q " + ques;    //node se chuyen thanh cau hoi moi duoc nguoi choi nhap
    temp->left = newQA("#A " + ans);    // con vat dung cho cau hoi moi
}
```

4. bool play(QA* root)

```
bool play(QA* root) // Tra ve true neu doan dung va false neu khong doan duoc con vat
{
    char ans='y';
    QA* temp=root;
    vector<string> vans;
    string sen;
    while (root)//Chay den het Node cau tra loi #A
    {
        cout << root->sen.substr(3, root->sen.length() - 3) << " (y/n): "; // xuat ra cau hoi bo phan dau (#Q, #A)
        cin >> ans;
        temp = root;
        if (ans == 'y')
        {
            sen = root->sen.substr(3, root->sen.length() - 3) + " YES\n"; // Luu lai cau hoi va cau tra loi YES cua nguoi choi
            vans.push_back(sen);
            root = root->left;
        }
        else
        {
            sen = root->sen.substr(3, root->sen.length() - 3) + " NO\n"; //Luu lai cau hoi va cau tra loi No cua nguoi choi
            vans.push_back(sen);
            root = root->right;
        }
    }
    if (ans == 'y') //Doan dung
    {
        cout << "I wont! I guessed it!\n";
        return true;
    }
    update(temp, vans);//Neu khong doan duoc thi them con vat moi
    return false;
}
```

5. void save(QA* root)

```
void save(QA* root)// Luu lai cay BST vao file theo cau truc NLR
{
    fs << root->sen << endl;
    if (root->left) save(root->left);
    if (root->right) save(root->right);
    return;
}
```

III. Khó khăn:

Em không gặp phải khó khăn nào quá lớn. Một chút khó khăn ở cách tổ chức chương trình sao cho sạch đẹp nhưng em đã tìm ra giải pháp rất nhanh sau đó.