

Devoir Final de Cpp

Nom: Charraj
Prenom: Zakariaa
Filier: IDAI

```
#include <iostream>
```

```
class ElPile {  
public:  
    // Liste simplement chainee  
    int valeur;  
    ElPile* suivant;  
  
    ElPile(int val) : valeur(val), suivant(nullptr) {}  
};
```

```
class Stack {  
private:  
    ElPile* pile;  
    // le variable n vas augmenter chaque fois la fonction operator<<() est appelee jusqu'a la liste est pleine.  
    static int n;
```

```
public:  
    Stack() : pile(nullptr) {  
        // L'initialisation de la taille de la pile.  
        n = 1;  
    }
```

```
    ~Stack() {  
        while (!est_vide()) {  
            pop();  
        }  
    }
```

```
    void operator<<(int valeur) {  
        if (n!=20) {
```

```

    ElPile* nouvelElement = new ElPile(valeur);
    nouvelElement->suivant = pile;
    pile = nouvelElement;
    n+=1;
}
}

void operator>>() {
    if (!est_vide()) {
        ElPile* temp = pile;
        pile = pile->suivant;
        delete temp;
        n-=1;
    }
    // Gerer le cas la pile est vide.
    std::cout << "Pile est vide!!" << std::endl;
}

int operator++() {

}

// Opérateur --
int operator--() {
    return pile == 0;
}

bool est_vide() {
    return pile == nullptr;
}
};

int main() {
    // La creation des piles.
    Stack n1, n2, n3, p;

    // remplir les pile pour executer les operations.

    p << n1 << n2 << n3 ;
    p >> n1 >> n2 << n3 ;

    // free up tous les pile.
    while (!n1.estVide()) {
        int valeur = n1.pop();
        std::cout << "n1 : " << valeur << std::endl;
    }

    while (!n2.estVide()) {

```

```
    int valeur = n2.pop();
    std::cout << "n2 : " << valeur << std::endl;
}

while (!n3.estVide()) {
    int valeur = n3.pop();
    std::cout << "n3 : " << valeur << std::endl;
}

while (!n4.estVide()) {
    int valeur = n4.pop();
    std::cout << "n4 : " << valeur << std::endl;
}

return 0;
}
~
```