Министерство высшего образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«Пермский национальный исследовательский политехнический университет» (ПНИПУ)**

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

ОТЧЁТ

по лабораторной работе №8

Тема: «Программа, управляемая событиями»

Выполнил

Студент группы РИС-22-2б

Арапов М.В.

Проверил доц. Кафедры ИТАС

Полякова О. А.

Пермь 2023

# Постановка задачи

Базовый класс:

ЧЕЛОВЕК (Person)

Имя - string

Возраст - int

Производный класс

СОТРУДНИК (Employee)

Заработная плата - float

Должность - string

Группа - Вектор (Vector).

Команды:

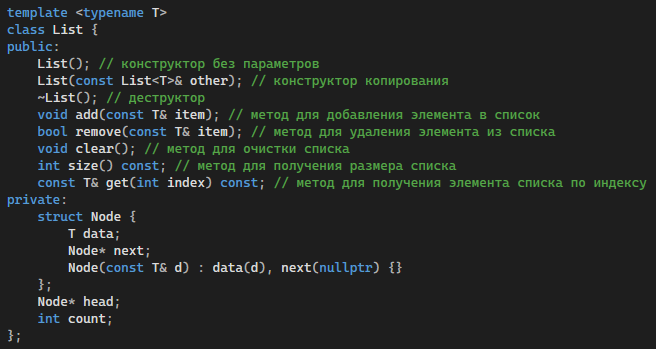
* Создать группу (формат команды: m количество элементов группы).
* Добавить элемент в группу (формат команды: +)
* Удалить элемент из группы (формат команды -)
* Вывести информацию об элементах группы (формат команды: s)
* Вывести информацию об имени элемента с номером k (формат команды : z k, где k - целое число)
* Конец работы (формат команды: q)

**Контрольные вопросы**

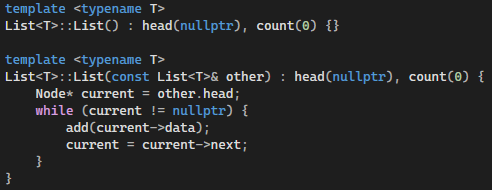
1. Что такое класс-группа? Привести примеры таких классов.

Класс-группа - это класс, который представляет собой группу объектов одного типа. Примерами таких классов могут быть классы "Массив", "Очередь", "Стек", "Список" и т.д.

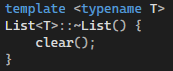
1. Привести пример описания класса-группы Список (List).



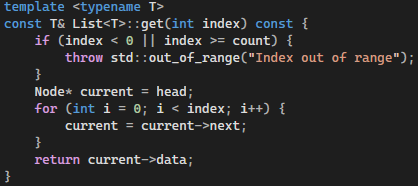
1. Привести пример конструктора (с параметром, без параметров, копирования) для класса-группы Список.



1. Привести пример деструктора для класса-группы Список.



1. Привести пример метода для просмотра элементов для класса-группы Список.



1. Какой вид иерархии дает группа?

Группа объектов может давать иерархию, например, если у нас есть группа объектов "Фигуры", то мы можем создать подклассы "Круг", "Прямоугольник", "Треугольник" и т.д., которые будут наследоваться от базового класса "Фигура".

1. Почему во главе иерархии классов, содержащихся в группе объектов должен находится абстрактный класс?

Абстрактный класс должен быть во главе иерархии классов, содержащихся в группе объектов, потому что он определяет общий интерфейс для всех подклассов и не может быть создан сам по себе. Это позволяет упростить работу с группой объектов и использовать полиморфизм для работы с различными типами объектов.

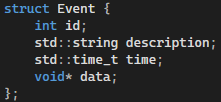
1. Что такое событие? Для чего используются события?

Событие - это сообщение о том, что что-то произошло в программе или в системе. События используются для уведомления других частей программы о произошедших изменениях или событиях, например, о нажатии кнопки пользователем или о завершении операции.

1. Какие характеристики должно иметь событие-сообщение?

Событие-сообщение должно иметь уникальный идентификатор, описание события, данные, связанные с событием (если есть) и время, когда событие произошло.

1. Привести пример структуры, описывающей событие.



1. Полю what присваиваются значения, которые определяют тип события. Например:

0 – событие связано с мышью;

1 – событие связано с нажатием клавиши;

2 – событие связано с сообщением.

1. Полю command присваиваются значения, которые определяют код команды. Например:

0 – команда открытия файла;

1 – команда сохранения файла;

2 – команда печати документа.

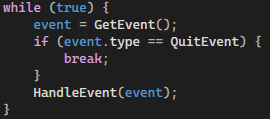
Значения полей message и a используются для передачи параметров команды.

1. Поля "a" и "message" используются как параметры команды, которые могут быть переданы в структуру Kevent. Например, если тип события "what" указывает на выполнение определенной команды, то код команды может быть записан в поле "command", а ее параметры могут быть переданы в поля "message" и "a". Таким образом, эти поля используются для передачи дополнительной информации, необходимой для выполнения определенной команды.
2. Какие методы необходимы для организации обработки сообщений?

Для организации обработки сообщений необходимы методы, обрабатывающие конкретные типы событий, например, OnClick для обработки нажатия на кнопку, OnKeyDown для обработки нажатия клавиши и т.д.

1. Какой вид имеет главный цикл обработки событий-сообщений?

Главный цикл обработки событий-сообщений имеет вид:



1. Какую функцию выполняет метод ClearEvent()? Каким образом?

Метод ClearEvent() очищает содержимое структуры TEvent, устанавливая все ее поля в значения по умолчанию.

1. Какую функцию выполняет метод HandleEvent ()?Каким образом?

Метод HandleEvent() выполняет обработку конкретного события в зависимости от его типа. Для этого он вызывает соответствующий метод, например, OnClick для обработки нажатия на кнопку, OnKeyDown для обработки нажатия клавиши и т.д.

1. Какую функцию выполняет метод GetEvent ()?

Метод GetEvent() получает следующее событие из очереди сообщений и возвращает его в виде структуры TEvent.

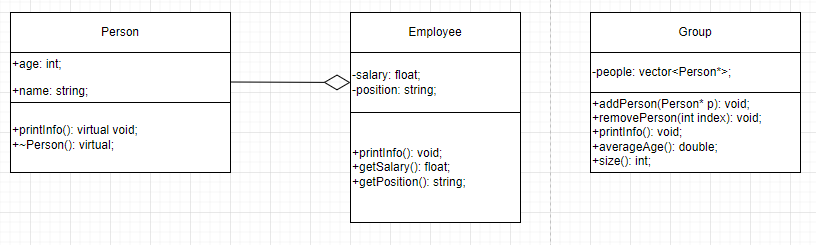
1. Для чего используется поле End State? Какой класс (объект) содержит это поле?

Поле End State используется для указания состояния приложения после завершения работы главного цикла обработки событий-сообщений. Это поле содержится в классе TApplication.

1. Для чего используется функция Valid()?

Функция Valid() используется для проверки корректности значений полей структуры TEvent. Она возвращает true, если значения полей находятся в допустимых пределах, и false в противном случае.

UML таблица



Код программы

#include <iostream>

#include <string>

#include <vector>

using namespace std;

// Базовый класс

class Person {

public:

Person(string name, int age) : name(name), age(age) {}

virtual ~Person() {}

virtual void printInfo() const = 0; // чисто виртуальный метод

public:

string name;

int age;

};

// Производный класс

class Employee : public Person {

public:

Employee(string name, int age, float salary, string position)

: Person(name, age), salary(salary), position(position) {}

void printInfo() const override {

cout << "Employee: " << name << ", Age: " << age << ", Salary: " << salary << ", Position: " << position << endl;

}

float getSalary() const {

return salary;

}

string getPosition() const {

return position;

}

private:

float salary;

string position;

};

// Класс-группа

class Group {

public:

Group() {}

~Group() {

for (auto& p : people) {

delete p;

}

}

void addPerson(Person\* p) {

people.push\_back(p);

}

void removePerson(int index) {

delete people[index];

people.erase(people.begin() + index);

}

void printInfo() const {

for (const auto& p : people) {

p->printInfo();

}

}

double averageAge() const {

double sum = 0;

for (const auto& p : people) {

sum += p->age;

}

return sum / people.size();

}

int size() const {

return people.size();

}

Person\* getPerson(int index) {

return people[index];

}

private:

vector<Person\*> people;

};

int main() {

Group group;

while (true) {

char command;

cout << "Enter command (+ to add, - to remove, s to show, z to show by index, q to quit): ";

cin >> command;

if (command == '+') {

string name, position;

int age;

float salary;

cout << "Enter name: ";

cin >> name;

cout << "Enter age: ";

cin >> age;

cout << "Enter salary: ";

cin >> salary;

cout << "Enter position: ";

cin >> position;

group.addPerson(new Employee(name, age, salary, position));

}

else if (command == '-') {

int index;

cout << "Enter index to remove: ";

cin >> index;

if (index >= 0 && index < group.size()) {

group.removePerson(index);

}

else {

cout << "Index is out of range!" << endl;

}

}

else if (command == 's') {

group.printInfo();

}

else if (command == 'z') {

int index;

cout << "Enter index to show: ";

cin >> index;

if (index >= 0 && index < group.size()) {

group.getPerson(index)->printInfo();

}

else {

cout << "Index is out of range!" << endl;

}

}

else if (command == 'q') {

break;

}

else {

cout << "Unknown command!" << endl;

}

}

return 0;

}