Министерство высшего образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

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

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

ОТЧЁТ

Тема: «Лабораторная работа №8»

Выполнил

Студент группы РИС-22-2б

Прядеин И.А.

Проверил доцент кафедры

ИТАС

Полякова О. А.

Пермь 2023

Постановка задачи

1. Определить иерархию пользовательских классов. Во главе иерархии должен стоять абстрактный класс с чисто виртуальными методами ввода и вывода информации об атрибутах объектов.
2. Реализовать конструкторы, деструктор, операцию присваивания, селекторы и модификаторы.
3. Определить класс-группу на основе структуры, указанной в варианте.
4. Для группы реализовать конструкторы, деструктор, методы для добавления и удаления элементов в группу, метод просмотра группы, перегрузить операцию для получения информации о размере группы.
5. Определить класс Диалог - наследника группы, в котором реализовать методы для обработки событий.
6. Добавить методы для обработки событий группой и объектами пользовательских классов.
7. Написать тестирующую программу.
8. Нарисовать диаграмму классов и диаграмму объектов.

**Вариант 8:**

Базовый класс:

ЧЕЛОВЕК (Person)

Имя - string

Возраст - int

Производный класс

АБИТУРИЕНТ (ABITURIENT)

Количество баллов - int

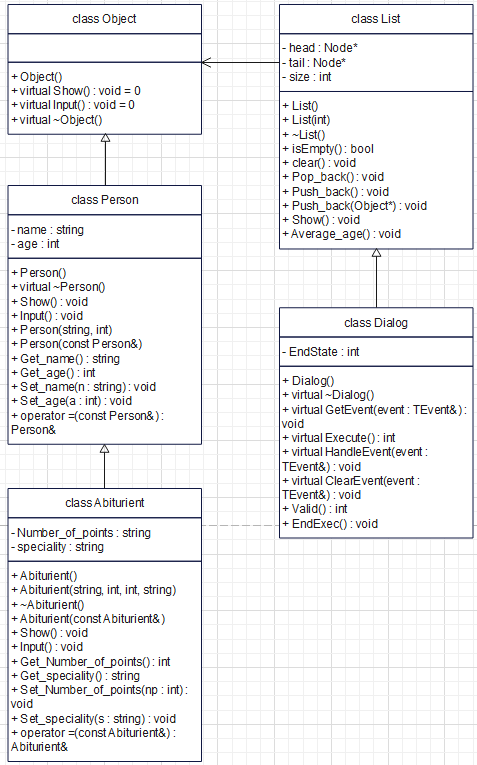
Специальность - string

Группа - Список (List).

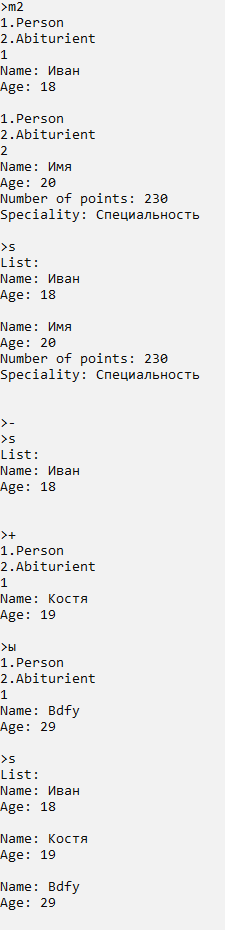
Комманды:

* 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 о среднем возрасте (формат команды: z)
* Конец работы (формат команды: q)

**UML диаграмма:**



**Скриншот результата выполнения программы**



**Ответы на вопросы:**

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

**Ответ:** Класс-группа - это объект, в который включены другие объекты. Объекты, входящие в группу называются элементами группы. Элементы группы, в свою очередь, могут быть группой.

Примеры групп:

1. Окно в интерактивной программе, которое владеет такими элементами, как поля ввода и редактирования данных, кнопки, списки выбора и т.д.
2. Некая огранизвационная структура.
3. Привести пример описания класса-группы Список (List).

**Ответ:**

struct Node {

Object\* data;

Node\* next;

Node\* prev;

Node (Object\* data, Node\* next = nullptr, Node\* prev = nullptr) {

this->data = data;

this->next = next;

this->prev = prev;

}

};

class List {

private:

Node\* head;

Node\* tail;

int size;

public:

List();

List(int);

~List();

bool isEmpty();

void clear();

void Pop\_back();

void Push\_back();

void Push\_back(Object\*);

void Show();

void Average\_age();

};

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

List() { // Конструктор без параметров

head = nullptr;

tail = nullptr;

size = 0;

}

List(int size) { // Конструктор с параметрами

if (size < 0)

throw invalid\_argument("Size cannot be negative");

head = nullptr;

tail = nullptr;

this->size = 0;

for (int i = 0; i < size; i++)

Push\_back();

}

List(const List& other) { // Конструктор копирования

Node\* current = head;

while (head != nullptr) {

other.Push\_back(current);

current = current->next;

}

}

1. Привести пример деструктора для класса-группы Список.

**Овтет:**

~List() {

clear();

}

void List::clear() {

while (!isEmpty())

Pop\_back();

}

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

**Ответ:**

void List::Show() {

Node\* current = head;

cout << "List:\n";

while (current != nullptr) {

Object\* p = current->data;

p->Show();

cout << endl;

current = current->next;

}

cout << "\n";

}

1. Какой вид иерархии даёт группа?

**Ответ:** Группа даёт второй вид иерархии - иерархию объектов (иерархию типа целое/часть), построенную на основе агрегации, первый вид - иерархия классов, построенная на основе наследования.

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

**Ответ:** Для того, чтобы можно было наследовать от него другие элементы. содержащиеся в группе.

1. Что такое событие? Для чего используются события?

**Ответ:** Событие - пакет информации, создаваемый объектно-ориентированной средой в ответ на те или иные действия пользователя, которым обмениваются объекты.

В ответ на событие, могут создаваться новые объекты, модифицироваться или уничтожаться существующие, что и приводит к изменению состояния программы.

1. Какие характеристики должно иметь событие-сообщение?

**Ответ:** Событие-сообщение, как правило, несёт следующие характеристики:

* код класса сообщения, отличающий сообщения объектов одного класса от объектов другого;
* адрес объекта, которому предназначено сообщение;
* информационное поле.

1. Привести пример структуры, описывающей событие.

**Ответ:**

struct TEvent {

int what;

union {

int command;

struct {

int message;

int a;

};

};

};

1. Задана структура события

struct TEvent {

int what;

union {

MouseEventType mouse;

KeyDownEvent keyDown;

MessageEvent message;

}

};

Какие значения, и в каких случаях присваиваются полю what?

**Ответ:** Полю what присваивается значения, которые содержат информацию о действиях мыши.

1. Задана структура события

struct TEvent {

int what;

union {

int command;

struct {

int message;

int a;

};

};

};

Какие значения, и в каких случаях присваиваются полю command?

**Ответ:** Поле command - код команды, которую необходимо выполнить при появлении события. В данном случае поле содержит параметр команды и код сообщения.

1. Задана структура события

struct TEvent {

int what;

union {

int command;

struct {

int message;

int a;

};

};

};

Для чего используется поля a и message?

**Ответ:** a - параметр команды, message - код сообщения.

1. Какие методы необходимы для организации обработки сообщений?

**Ответ:** Для организации обработки сообщений необходимы следующие методы:

* GetEvent
* Execute
* HandleEvent
* ClearEvent

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

**Ответ:**

void TDerivedClass::HandleEvent(TEvent& event) {

TBaseClass::handleEvent(event);

if (event.what == evCommand) {

switch (event.message.command) {

case cmCommand1:

ClearEvent(event);

break;

case cmCommand2:

ClearEvent(event);

break;

…

case cmCommandN:

ClearEvent(event);

break;

default:

break;

}

};

}

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

**Ответ:** ClearEvent очищает событие, когда оно отработано, что не обрабатывалось далее. ClearEvent очищает событие, присваивая полю event.what значение evNothing.

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

**Ответ:** HandleEvent - обработчик событий. Обрабатывает каждое событие нужным для него образом. Если объект должен обрабатывать определённое событие (сообщение), то его метод HandleEvent должен распознавать это событие и реагировать на него должным образом. Событие может распознаваться, например, по коду команды.

1. Какую функцию выполняет метод GetEvent()?

**Ответ:** GetEvent формирует события.

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

**Ответ:** Значение этой переменной проверяет метод-функция Valid, возвращающая значение true, если “Конец работы”.

1. Для чего используется функция Valid()?

**Ответ:** Метод Valid() служит для проверки значения переменной EndState, служащей для завершения работы программы.