Учреждение образования Федерации профсоюзов Беларуси

«Международный университет «МИТСО»

Рег. № \_\_\_\_\_\_\_\_\_\_ Кафедра информационных систем и технологий

Дата \_\_\_\_\_\_\_\_\_2020

**КУРСОВАЯ РАБОТА**

на тему \_\_\_\_Название\_\_\_Разработать программный модуль «Картотека недвижимости»\_\_\_\_\_\_\_\_

по дисциплине \_Название\_\_\_\_Основы конструирования программ\_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
| Основные замечания: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Отметка о допуске курсовой работы к  защите:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Дата: \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_\_  Подпись научного руководителя:  \_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Студент: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись)  Урбанович Владислав Сергеевич (Ф.И.О. – полностью)  Курс \_\_1\_\_, группа \_\_\_1920\_\_\_\_  Факультет \_\_экономический\_\_\_  Специальность \_\_Информационные системы и технологии\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Научный руководитель:  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (должность, ученая степень, ученое звание)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (Ф.И.О. – полностью) |

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc70868457)

[1.ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ 4](#_Toc70868458)

[**1.1. Общие положения 4**](#_Toc70868459)

[**1.2. Постановка задачи 4**](#_Toc70868460)

[**1.3. Спецификация входных и выходных данных 4**](#_Toc70868461)

[**1.4. Выбор средства 5**](#_Toc70868462)

[**1.5.Описывается требуемое оборудование и программное обеспечение 6**](#_Toc70868463)

[2.ТЕХНОЛОГИЯ РАЗРАБОТКИ ПРИЛОЖЕНИЯ 7](#_Toc70868464)

[**2.1. Алгоритм решения 7**](#_Toc70868465)

[**2.2. Описание программы 8**](#_Toc70868466)

[**2.3. Результаты работы программы 11**](#_Toc70868467)

[3.РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ 12](#_Toc70868468)

[ЗАКЛЮЧЕНИЕ 13](#_Toc70868469)

[СПИСОК ЛИТЕРАТУРЫ 14](#_Toc70868470)

[ПРИЛОЖЕНИЕ 15](#_Toc70868471)

# ВВЕДЕНИЕ

Была поставлена задача разработать программный продукт «Информационные процессы компьютерной фирмы (разработка ПО)», предназначенный для использования в IT компании. В программном продукте необходимо реализовать автоматизацию подписания контракта.

Программа создана для автоматизации подписания контракта как со стороны заказчика, как и со стороны исполнителя. Что позволит сильно сократить время переговоров.

Цель разработки обеспечить работу программного продукта быстро и максимально эффективно, удовлетворять запросы клиента и заказчика, а также должен быть возможность пополнять базу данных, для учёта подписанных контрактов. Основная же задача стоит в высокой оптимизация, чтобы пользователь мог запустить даже на самом слабом устройстве.

Задачи данного программного продукта: быстро и гибко редактироваться со стороны владельцев компании. С точки зрения использования пользователя программа должна иметь красивый, понятный и интуитивный интерфейс чтобы было клиенту приятно пользоваться программой. Так же требуется предусмотреть возможность добавления новых функций в программе.

Для разработки программы был использован текстовый файл и язык программирования C++ разработанный в среде Visual Studio компаний Microsoft.

Visual Studio-линейка продуктов компании Microsoft, включающих интегрированную среду разработки программного обеспечения и ряд других инструментальных средств. Данные продукты позволяют разрабатывать как консольные приложения, так и приложения с графическим интерфейсом, в том числе с поддержкой Windows Forms, а также веб-сайты, веб-приложения, веб-службы, как в родном, так и в управляемом кодах для всех платформ, поддерживающих Windows, Windows Mobile, Windows Ce, .Net Framework, Xbox, Windows Phone .Net Compact Framework и Silverling.

# 1.ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ

## 1.1. Общие положения

В современном бизнесе сложилась ситуация, при которой компании необходимо структурировать, обрабатывать, структурировать и катализировать информацию с большим количеством различного рода информации. Для обеспечение подобного рода операций необходимо использовать технологии, которые будут обеспечивать возможность подобной работы.

Была поставлена задача разработать программный продукт «Информационные процессы компьютерной фирмы (разработка ПО)», предназначенный для использования в IT компании. В программном продукте необходимо реализовать автоматизацию подписания контракта.

Фирма разрабатывает два вида программного обеспечения – заказное и коробочное. В первом случае находиться заказчик (предприятия), с ним заключается договор на разработку, который затем выполняется. Коробочный программный продукт разрабатывается следующим образом. Прорабатывается рынок и определяется потребность в едином программном обеспечении для многих организаций. Далее ищется источник финансирования (собственные либо инвесторы) и на своей страх и риск программа разрабатывается. Программа разрабатывается вместе с инструкцией (“коробка”) и поступает на рынок на продажу

Данный программный продукт выгодно отличается от других подобных тем что, пользуясь им, пользователь может не хранить данные заказчика, но и добавлять их.

## 1.2. Постановка задачи

Исходя из задач, описанных выше главными требованиями к языку, на котором будет написана программы — это тонкая настройка и возможность использованием минимум вычислительных ресурсов добиваться максимум результата.

## 1.3. Спецификация входных и выходных данных

В данном программном продукте используются следующие типы данных типа данных: sting, double, int bool. А также присутствуют пользовательские типы данных: Developer, Team-Developer, Customer, Contract.

boll – используется только внутри программы, для проверки, подсчёта, иннерваций и так далее. Пользователь на данный этого типа влиять и пользоваться не может.

String – используется в двух случаях. Либо для наименования данных (присвоение данным название удобного пользователя). Во втором же случае данный тип данных используется для вывода всех данных на экран.

Int и double – используется для реализации цены за конкретный проект.

В данном проекте реализовано большое количество классов. Это необходимо для того, чтобы разделить проект на блоки. Чтобы не множить сущности и код был более читаемым и красивым.

* Developer – данный класс содержит информацию об разработчике, его имя и фамилия, а также коллекция, в которой содержаться технологии, которыми владеет разработчик.
* Team-Developer – связан с классам Developer. В данном классе содержаться разработчики, которые собраны в единую команду которые будут работать над проектом.
* Customer – в данном классе содержаться информация о заказчике, его имя и фамилия, а так же его финансовое состояние.
* Contract – это основной класс в котором используется все предыдущие. В нём происходят все взаимодействия с предыдущими классами. Так же в данном классе содержаться данные о самом проекте.

## 1.4. Выбор средства

Для выполнения данных задач очень подходит такой язык программирования как с++. Так как его из-за того, что он является весьма низкоуровневым языком программирования, а значит он больше других приближен к машинному коду, из-за этого с++ дает возможность оптимизировать проект так что даже на очень старых устройствах он будет работать.

Среда разработки Microsoft Visual Studio. Вид приложения – консольное. Программа построена на базе объектно-ориентированного программирования. Способ организации данных поля классов. Способ хранения данных – динамические массивы. Каждая логическая завершенная данных подзадача программы реализована в виде методов. Построение программного кода соответствует с <<C++ Code Convention>>. К защите курсовой предоставляется: консольное приложение и пояснительная записка

Windows 10 – операционная система для персональных компьютеров и рабочих станций, разработанная корпорацией Microsoft в рамках семейства Windows NT. После Windows 8.1 система получила номер 10, минуя 9. Серверный аналог Windows 10 – Windows Server 2016. Система призвана стать единой для разных устройств, таких как персональные компьютеры, планшеты, смартфоны, консоли Xbox One и пр. Доступна единая платформа разработки и единый магазин универсальных приложений, совместимых со всеми поддерживаемыми устройствами. Windows 10 поставляется в качестве услуги с выпуском обновлений на протяжении всего цикла поддержки. В течение первого года после выхода системы пользователи могли бесплатно обновиться до Windows 10 на устройствах под управлением лицензионных копий Windows 7, Windows 8.1 и Windows Phone 8.1. Среди значимых нововведений – голосовая помощница Кортана, возможность создания и переключения нескольких рабочих столов и др. Windows 10 – последняя «коробочная» версия Windows, все последующие версии будут распространяться исключительно в цифровом виде.

## 1.5.Описывается требуемое оборудование и программное обеспечение

Для полноценного функционирования разработанного программного приложения, качественного выполнения всех поставленных задач, требуется ПК следующей конфигурации:

* процессор Intel Core i3;
* оперативная память DDR3 512 МБ или больше;
* минимально свободного места на жёстком диске 512 МБ;
* мышь;
* операционная система Windows 7/8/10.

# 2.ТЕХНОЛОГИЯ РАЗРАБОТКИ ПРИЛОЖЕНИЯ

## 2.1. Алгоритм решения

В текущей главе представлено описание алгоритма, который имеет начало алгоритма и конец, внутри алгоритма имеются циклы и иные операций.

Так как в качестве языка разработке выбран C++. Он имеет характерную особенность. Весть алгоритм работы программы реализуется в функции main.

Что же касается непосредственно данного проекта. То функция main содержит крайне мало кода. Это получилось за счёт вынесения функционала проекта в методы класса, а также за счёт разбивки методов на составляющие. Эти операции повысили читаемость кода.

Вначале происходит добавление создание трёх команд разработчиков и последующие присвоение к каждой команде определённый разработчиков (рис 2.1)

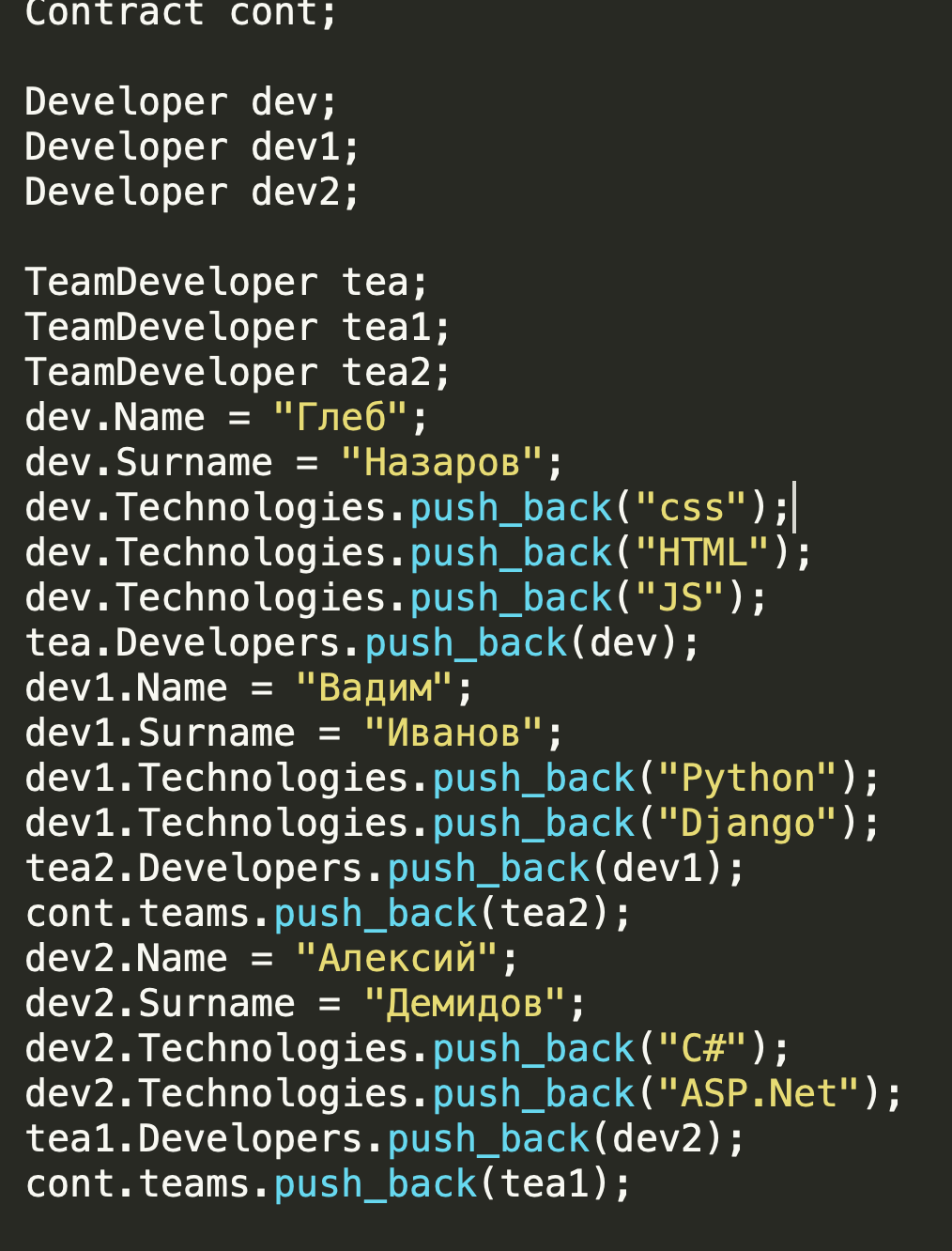


Рис 2.1

Далее происходит работа с классом Contract (рис 2.2)

Метод **SubscribeContract –** данный метода является очень сложный алгоритмом и состоит из нескольким методом. Подробнее о нём поговорим в следующей главе.

Суть же данного элемента состоит в том что происходит взаимодействие с каждым классом и присвоение каждому классу соответствующих данных, путём ввода данных в консоль.

Метод **Print** – данный метод выводит всё содержимое контракта (разработчики, цена проекта, данные заказчика, дата заключения контракта, информация о проекте).

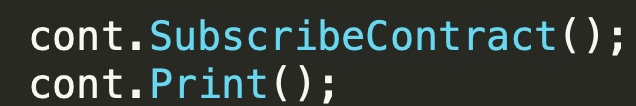


Рис 2.2.

## 2.2. Описание программы

В данной программы используются следующие библиотеки:

#include <iostream> - основная библиотека, которая реализует основные возможность языка С++.

#include <string> - библиотека для возможности работы с типом данных string.

#include <random> - данная библиотека необходима для реализации работы с генератором случайных числах.

#include <ctime> - данная библиотека необходима для реализации работы с датой и времени.

#include <deque> - данная библиотека необходима для реализации работы с двухсторонней очередью.

В данном программном продукте используется 4 класса Developer, TeamDeveloper, Customer, Contract.

**Developer** (рис 2.3.)**:**

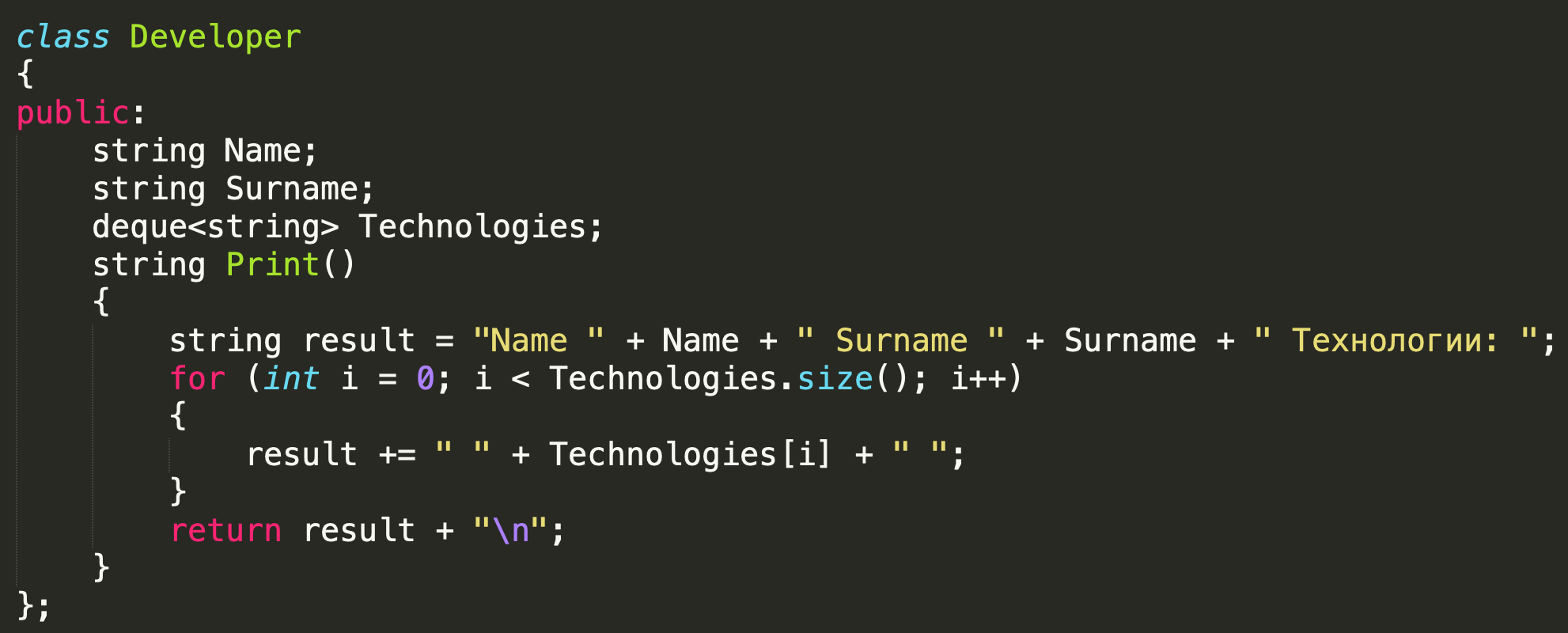
Данный класс содержит три переменные:

Name – имя разработчика.

Surname – фамилия разработчика.

Technologies – технологии которыми владеет разработчик.

Метод string **Print** – возвращает строковое значение всех полей класса, необходим для вывода данных на консоль.



Класс **TeamDeveloper** (рис 2.4.):

Данный класс напрямую связан с приводящим так как хранит только одно поля, которое содержит коллекцию разработчиков. Данный метод представляет собой команду разработчиков, которая потом вписывается контракт

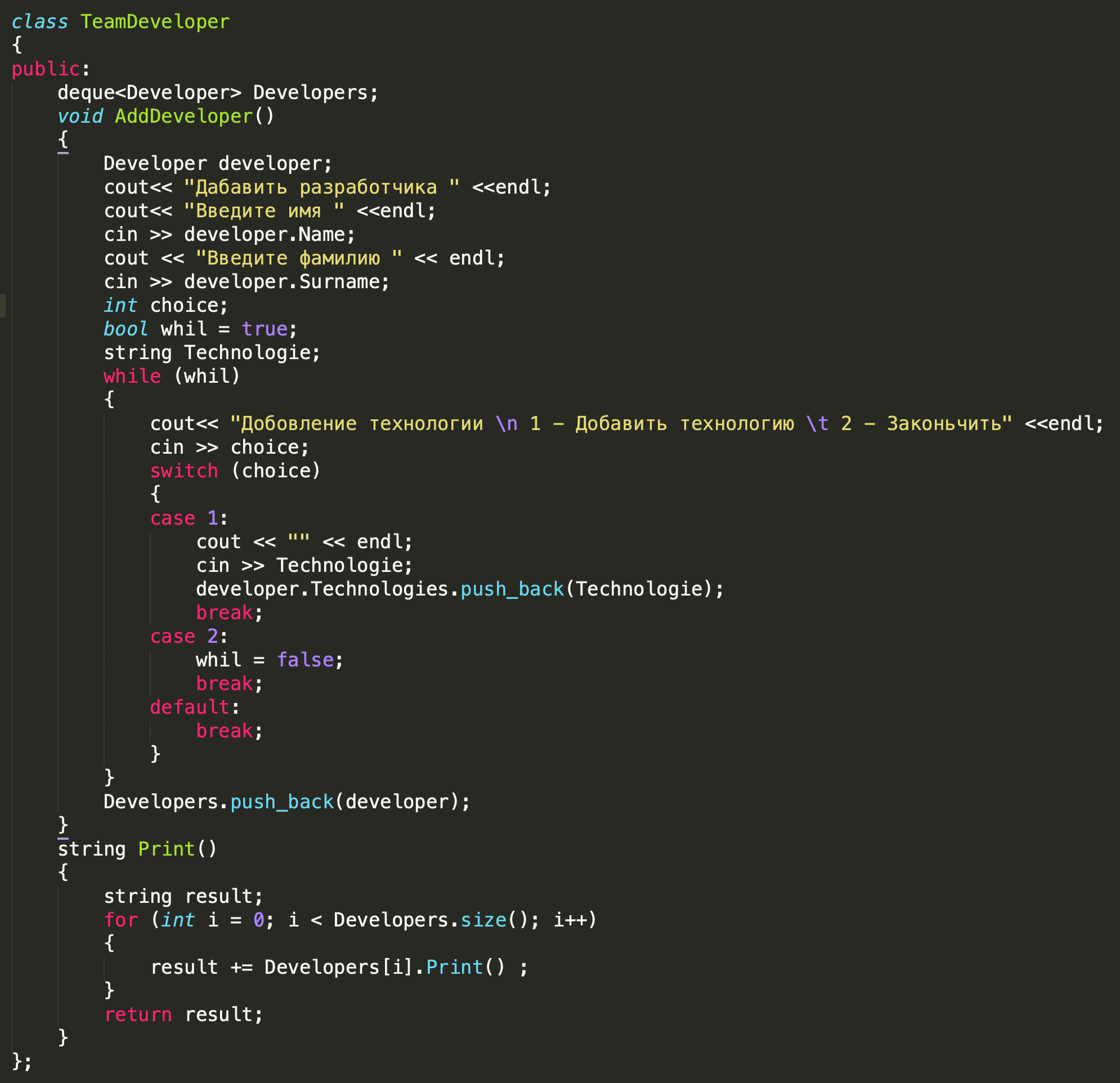


Рис (2.5)

Метод **Print** - возвращает строковое значение всех полей класса, необходим для вывода данных на консоль. Данный метод так же взаимодействует с каждым разработчиком и вызывает у каждого разработчика одноимённый метод Print.

Метод **AddDeveloper**() – данный метод реализует алгоритм добавления в команду разработчиков нового программисту. Путём создания экземпляра класса Developer и последующей его инициализации. После этого программист добавляется к команде.

Класс **Customer**(рис 2.6.)

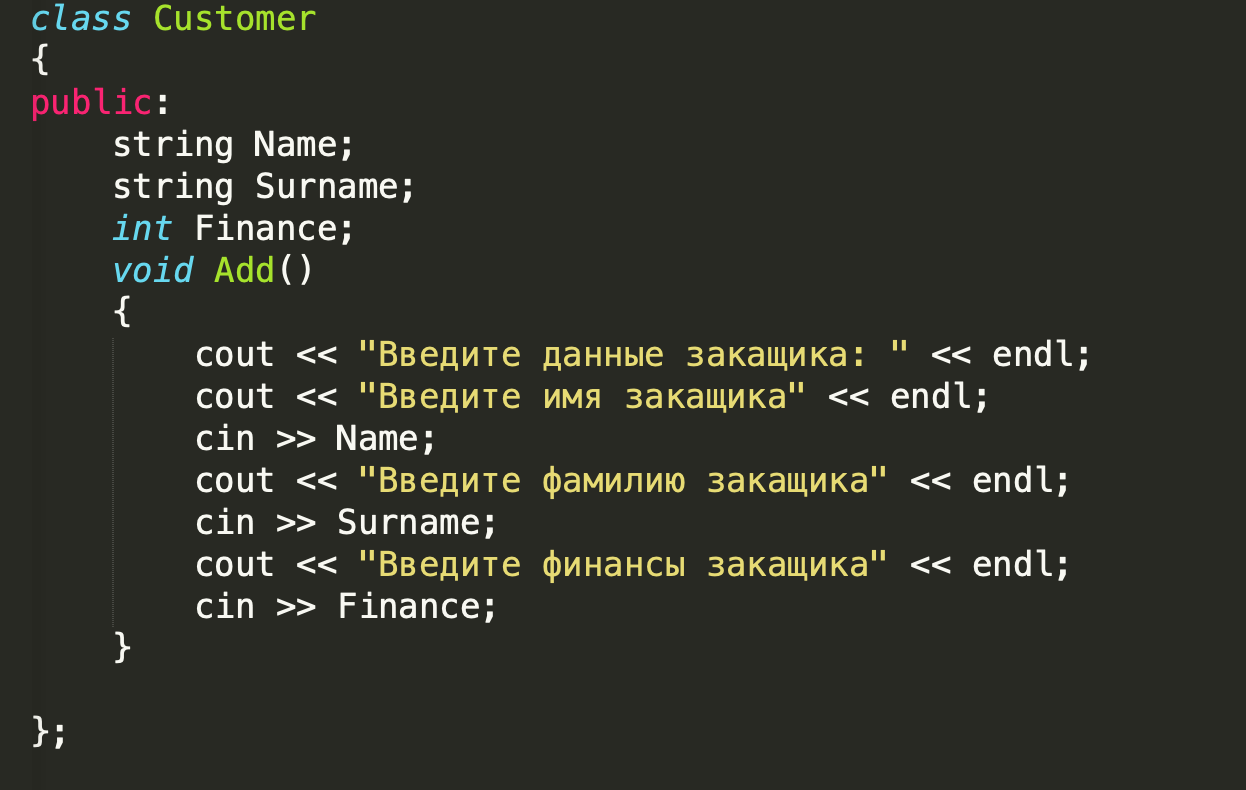
Данный класс содержит три переменные:

Name – имя заказчика.

Surname – фамилия заказчика.

Finance – количества денег у заказчика.

Метод **Add** – данный метод реализует инициализацию пред



Класс **Contract**(рис 2.7)

Данный класс содержит 5 переменные:

Customer – содержит данные о клиенте

Teams – содержит коллекцию всех команд которые возможно возможно подключить к проекту.

Team – содержит команду разработчиков которая будут работать над проектом.

Prise – содержит цену проекта которую должен будет необходимо оплатить заказчику.

Name – содержит название проекта



Рис 2.7

Методы:

**Add**() – данный метод позволяет сделать выбор команды разработчиков которая будет работать над проектом.

**AddDataContract**() – данный метод позволяет ввести данные проекта. Его цена и название проекта.

**SubscribeContract**() – Данный метод вызывает 2 предыдущих метода.

**Print**() – Выводит все данные заказа включая данные клиента, команда разработчиков, цена проекта, время заключения контракта.

## 2.3. Результаты работы программы

Результат работы программы приведён к единому методу класса Contract, Print (Рис 2.8).

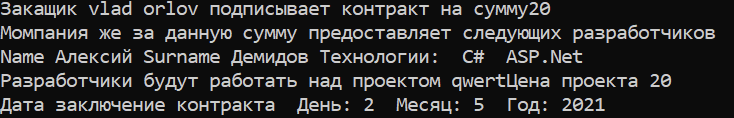


Рис 2.8

# 3.РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

Всё взаимодействие с данным программным продуктом происходит через взаимодействие с консолью, путям ввода данных в консоль (рис 3.1).

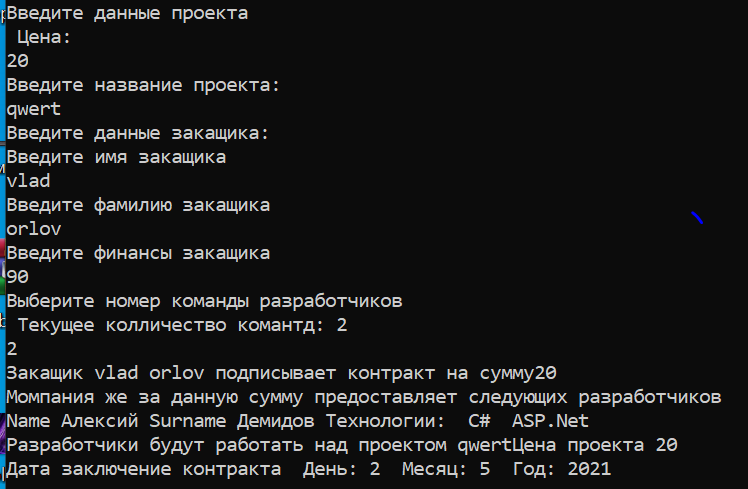


Рис 3.1

Вначале вводятся цена проекта, необходимо ввести целочисленное значение. Далее вводиться название проекта, которое представляет строковое значение.

Потом необходимо ввести данные заказчика, вначале вводятся имя и фамилия клиента, представленные строковым значением. Далее надо ввести финансы заказчика.

После этого необходимо ввести номер команды разработчика, которая будет работать с конкретным проектом. Для предотвращения ошибок в консоль выводиться количество команд разработчиков.

# ЗАКЛЮЧЕНИЕ

Исходя из поставленных целей и задач было сконструировано, разработана и протестирован программный модуль «Информационные процессы компьютерной фирмы (разработка ПО)», предназначенный для использования в IT компании. В программном продукте необходимо реализовать автоматизацию подписания контракта.

Были реализованы задачи:

1. Хранение данных заказов
2. Реализация присвоения проекта команде
3. Вывод данных
4. Оформление акта контракта

Цель была выполнены, программный продукт имеет следующие преимущества:

1. Работа программного продукта происходит быстро и максимально эффективно. Программа удовлетворяет запросам пользователя.
2. Имеет возможность пополнять базу данных, для расширения ассортимента товаров.
3. Так же программы имеет высокую оптимизацию. Из-за этого программа работает весьма быстра и запускается даже на самом слабом устройстве.

Так же данный программный продукт выгодно отличается от других представленных на рынке, возможностью не только покупать, но и продавать недвижимость, ещё она снабжена интуитивным интерфейс.

Таким образом программа работает без сбоев и все задачи были выполнены.

# СПИСОК ЛИТЕРАТУРЫ

1. Коплиен Дж. Мультипарадигменное проектирование для С++ / Коплиен Дж. – Питер, 2005.
2. Обзор обновлений и новых функций Windows 10 [Электронный ресурс]. – Microsoft, 2020. – Режим доступа: https://www.microsoft.com/ru-ru/windows/features. – Дата доступа: 22.05.2020.
3. Объектно-ориентированный анализ и проектирование с примерами приложений / Гради Буч [и др.]. – 3-е изд. – М.: ООО «И.Д. Вильямс», 2008. – 720 с.
4. Стивен Прата Язык программирования C++ / Стивен Прата – Вильямс, 2012.
5. Страуструп Б. Программирование. Принципы и практика с использованием C++ / Страуструп Б. – 2-е изд. – Вильямс, 2016.
6. Model-View-Presenter [Электронный ресурс]. – Википедия, 2020. – Режим доступа: https://ru.wikipedia.org/wiki/Model-View-Presenter. – Дата доступа: 22.04.2020.
7. SFML [Электронный ресурс]. – SFML, 2020. – Режим доступа: https://www.sfml-dev.org. – Дата доступа: 22.04.2020.

# ПРИЛОЖЕНИЕ

#include <iostream>

#include <string>

#include <deque>

#include <windows.h>

using namespace std;

class Developer

{

public:

string Name;

string Surname;

deque<string> Technologies;

string Print()

{

string result = "Name " + Name + " Surname " + Surname + " Teхнологии: ";

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

{

result += " " + Technologies[i] + " ";

}

return result + "\n";

}

};

class TeamDeveloper

{

public:

deque<Developer> Developers;

void AddDeveloper()

{

Developer developer;

cout<< "Дабавить разработчика " <<endl;

cout<< "Введите имя " <<endl;

cin >> developer.Name;

cout << "Введите фамилию " << endl;

cin >> developer.Surname;

int choice;

bool whil = true;

string Technologie;

while (whil)

{

cout<< "Добовление технологии \n 1 - Добавить технологию \t 2 - Законьчить" <<endl;

cin >> choice;

switch (choice)

{

case 1:

cout << "" << endl;

cin >> Technologie;

developer.Technologies.push\_back(Technologie);

break;

case 2:

whil = false;

break;

default:

break;

}

}

Developers.push\_back(developer);

}

string Print()

{

string result;

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

{

result += Developers[i].Print() ;

}

return result;

}

};

class Customer

{

public:

string Name;

string Surname;

int Finance;

void Add()

{

cout << "Введите данные закащика: " << endl;

cout << "Введите имя закащика" << endl;

cin >> Name;

cout << "Введите фамилию закащика" << endl;

cin >> Surname;

cout << "Введите финансы закащика" << endl;

cin >> Finance;

}

};

class Contract

{

public:

Customer customer;

deque <TeamDeveloper> teams;

TeamDeveloper team;

int Prise;

string Name;

TeamDeveloper Add()

{

cout << "Выберите номер команды разработчиков" << endl;

cout << " Текущее колличество комантд: " << teams.size() << endl;

int number;

cin >> number;

number--;

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

{

if (i == number)

{

return teams[i];

}

}

}

void AddDataContract()

{

cout << "Введите данные проекта \n Цена: " <<endl;

cin >> Prise;

cout << "Введите название проекта:" <<endl;

cin >> Name;

}

void SubscribeContract()

{

AddDataContract();

customer.Add();

team = Add();

}

void Print()

{

SYSTEMTIME st;

GetSystemTime(&st);

cout << "Закащик " << customer.Name << " " << customer.Surname << " подписывает контракт на сумму" << Prise << endl;

cout << "Момпания же за данную сумму предоставляет следующих разработчиков \n" << team.Print();

cout << "Разработчики будут работать над проектом " << Name << "Цена проекта " << Prise << endl;

cout << "Дата заключение контракта День: " << to\_string(st.wDay) << " Месяц: " << to\_string(st.wMonth) << " Год: " << to\_string(st.wYear) << endl;

}

};

int main()

{

setlocale(LC\_ALL, "Russian");

Contract cont;

Developer dev;

Developer dev1;

Developer dev2;

TeamDeveloper tea;

TeamDeveloper tea1;

TeamDeveloper tea2;

dev.Name = "Глеб";

dev.Surname = "Назаров";

dev.Technologies.push\_back("css");

dev.Technologies.push\_back("HTML");

dev.Technologies.push\_back("JS");

tea.Developers.push\_back(dev);

dev1.Name = "Вадим";

dev1.Surname = "Иванов";

dev1.Technologies.push\_back("Python");

dev1.Technologies.push\_back("Django");

tea2.Developers.push\_back(dev1);

cont.teams.push\_back(tea2);

dev2.Name = "Алексий";

dev2.Surname = "Демидов";

dev2.Technologies.push\_back("С#");

dev2.Technologies.push\_back("ASP.Net");

tea1.Developers.push\_back(dev2);

cont.teams.push\_back(tea1);

cont.SubscribeContract();

cont.Print();

}