

ИНФОРМАЦИОННЫЕ СИСТЕМЫ В СЕЙСМОСТОЙКОМ СТРОИТЕЛЬСТВЕ

Макалада имараттардын жер титироого туруктуулугун эсептоонун прикладдык программасынын пакетин кайра иштеп чыгаруунун ыкмасы келтирилет.

В статье излагается методика повторной разработки пакета прикладных программ (ППП) расчета на сейсмостойкость зданий.

The method of the development of the applied programs package of accounting to the seismic stability of the flexible buildings is stated in this article.

Программа была разработана в интегрированной среде разработки Delphi 7, далее последует описание данного программного изделия.

Прикладные программы, или приложения, Delphi создаются в интегрированной среде разработки. Пользовательский интерфейс этой среды служит для организации взаимодействия с программистом и включает в себя ряд окон, содержащих различные элементы управления. С помощью средств интегрированной среды разработчику удобно проектировать интерфейсную часть приложения, а также писать программный код и связывать его с элементами управления. В интегрированной среде разработки проходят все этапы создания приложения, включая отладку. Интегрированная среда разработки Delphi 7 представляет собой многооконную систему.

Язык Delphi реализует концепцию объектно-ориентированного программирования. Это означает, что функциональность приложения определяется набором взаимосвязанных задач, каждая из которых становится самостоятельным объектом. У объекта есть свойства (т.е. характеристики, или атрибуты), методы, определяющие его поведение, и события, на которые он реагирует. Одним из наиболее важных понятий объектно-ориентированного программирования является класс. Класс представляет собой дальнейшее развитие концепции типа и объединяет в себе задание не только структуры и размера переменных, но и выполняемых над ним операций. Объекты в программе всегда являются экземплярами того или иного класса.

Определение требований к программе – один из важнейших этапов. На этом этапе подробно описывается исходная информация и формулируются требования к результату. Кроме того, описывается поведение программы в особых случаях.

При разработке программы, предназначенной для работы в Windows, требования к ней могут включать пункты о желаемом виде диалоговых окон программы.

На этапе разработки алгоритма необходимо определить последовательность действий, которые надо выполнить для достижения поставленной цели, получения результата. Если задача может быть решена различными способами и, следовательно, возможны различные варианты алгоритма решения, то, используя некоторый критерий, выбираем наиболее подходящее решение.

При программировании в Delphi основная работа заключается в разработке подпрограмм. Задачу организации взаимодействия разработанных подпрограмм берет на себя Delphi. Также Delphi удобна для создания программ с визуальными компонентами, которые уже созданы в ней и имеют стандартный набор команд.

Поставленная задача заключалась в следующем: разработать пакет прикладных программ (ППП) вероятностных методов расчета на сейсмостойкость зданий. Необходимо создать программу, которая бы проводила тестирование на основе вероятностных методов расчета на сейсмостойкость зданий и сохраняла результат опроса для дальнейшего анализа. Программа должна помочь конструкторам и расчетчикам в проведении исследования сейсмостойкости в области строительства зданий и сооружений.

Среда разработки Delphi ориентирована, прежде всего, на создание программ для семейства операционных систем Windows. При этом большое внимание уделяется возможности визуальной разработки приложений с помощью большого набора готовых компонентов, а в стандартную поставку Delphi входят основные объекты, которые образуют удачно подобранную иерархию из 270 базовых классов, позволяющих избежать ручного кодирования. Эти компоненты охватывают практически все аспекты применения современных информационных технологий.

В процессе построения приложения необходимо выбирать из палитры компоненты, на основе которых будет строиться проект. Еще до компиляции видны результаты работы. В этом смысле проектирование в Delphi мало чем отличается от проектирования в интерпретирующей среде, однако после выполнения компиляции получаем код, который исполняется в 10-20 раз быстрее, чем то же самое, сделанное при помощи интерпретатора. Среда Delphi включает в себя полный набор визуальных инструментов для скоростной разработки приложений (RAD – rapid application development), поддерживающей разработку пользовательского интерфейса.

Расчет программы:

```
procedure TForm1.BitBtn1Click(Sender: TObject);
Var t,a:real; alf:real; g,p,m1,m2,m3,m4:real; sct,h,d,pi,j:real; e,s:real;
del1,del2,del3,del4:real; sd1,sd2,sd3,sd4:real; be1,be2,be3,be4:real;
dd1,dd2,dd3,dd4:real; eps1,eps2,eps3,eps4:real;
begin h:=StrToFloat(edit1.Text); t:=1/10; a:=0.2; alf:=a/sqr(t);
g:=10;p:=0.6; m1:=p/g; m2:=p/g; m3:=p/g; m4:=p/g; d:=0.4;
pi:=3.14; sct:=m1*alf; e:=2*1000000; j:=(pi*sqr(d))/64;
del1:=((h*h*h)/(12*e*j))*sct; del2:=((h*h*h)/(12*e*j))*sct;
del3:=((h*h*h)/(12*e*j))*sct; del4:=((h*h*h)/(12*e*j))*sct;
s:=(1+(del1/a))*sct; sd1:=(1+(del1/a))*sct;
be1:=1+(del1/a); sd2:=(1+(del2/del1))*sct;
sd3:=(1+(del2/del1))*sct; sd4:=(1+(del2/del1))*sct;
be2:=1+(del1/a); be3:=1+(del1/a); be4:=1+(del1/a); dd1:=((h*h*h)/(12*e*j))*sd1;
eps1:=((h*h*h)/(12*e*j))*sd1; dd2:=((h*h*h)/(12*e*j))*sd2;
dd3:=((h*h*h)/(12*e*j))*sd2; dd4:=((h*h*h)/(12*e*j))*sd2;
eps2:=eps1+dd2; eps3:=eps2+eps1+dd3;
eps4:=eps3+eps2+eps1+dd4; Label3.Caption:=FloatToStr(s);
Label6.Caption:=FloatToStr(eps1); Label8.Caption:=FloatToStr(eps2);
Label10.Caption:=FloatToStr(eps3); Label12.Caption:=FloatToStr(eps4); end
```

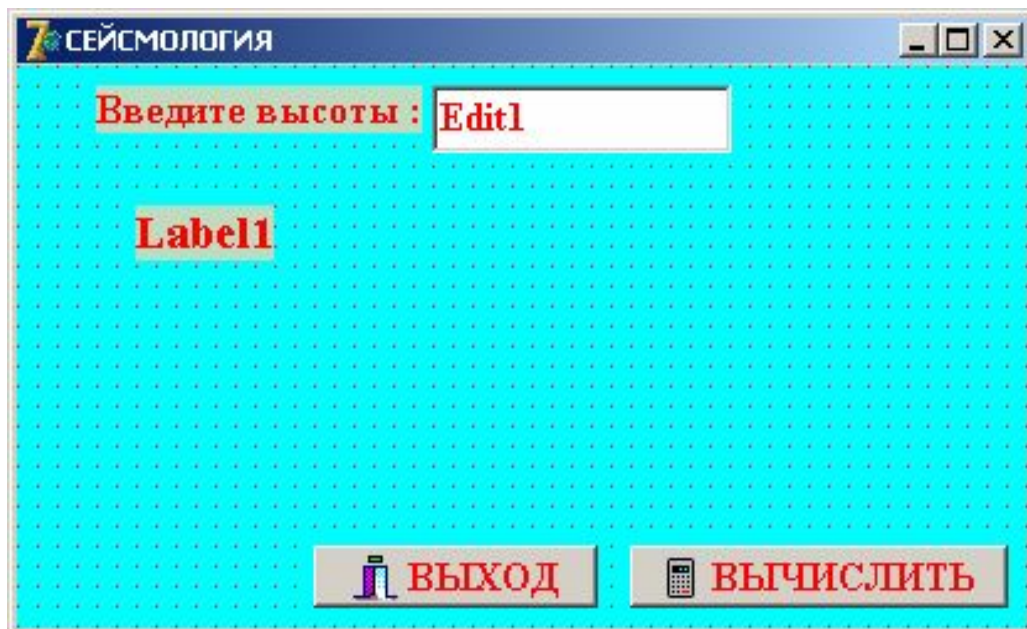
Диаграмма 1.1.



Диаграмма 1.2.



Диаграмма 1.3.



```
procedure TForm1.BitBtn1Click(Sender: TObject);  
var t,a1,all:real; m,h,si,e,j:real; s1,s2,s3:real; bet1,bet2,bet3:real;
```

```

begin h:=StrToFloat(edit1.Text); t:=1/10; e:=2*1000000; j:=0.00128;
a1:=0.1; m:=0.06; a1:=a1/sqr(t); s2:=m*a1; si:=((h*h*h)/(12*e*j))*s2;
s1:=(1-(si/a1))*s2; bet1:=(1-(si/si))*s2; s3:=(1-(si/a1))*s2; bet2:=1-(si/si);
s3:=(1-(si/si))*s2; bet3:=1-si/si;
Label1.Caption:='Dinamicheckay cila = '+FloatToStr(si)+#13+
'Dinamicheckay nagruski = '+FloatToStr(s2)+#13+
'Koefficient dinamicheckay cila = '+ FloatToStrf(bet1,ffGeneral,0,0)+#13+
'Koefficient dinamicheckay nagruski = '+ FloatToStrf(bet2,ffGeneral,0,0); end.

```

Организационная структура реализации алгоритма разработки программы схематически изображена на рис. 1.1.

Повторные вычисления программы построены на знании интерполяционного полинома Лагранжа (интерполяционный многочлен в форме Лагранжа). Именно с его помощью вычисляется значение функции при нескольких заданных пользователем значениях аргумента и некоторым другим входным данным. Интерполяционным многочленом называется многочлен степени не выше n , который в узлах принимает значения.

Условия применения полинома. Функция непрерывна на интервале и задана некоторыми своими значениями для соответствующих значений аргумента. Полином используется, когда необходимо найти значение этой функции в узле.

Созданная программа проводит тестирование на основе вероятностных методов расчета на сейсмостойкость зданий и сохраняет результат опроса для дальнейшего анализа. Программа помогает конструкторам и расчетчикам в проведении исследования сейсмостойкости в области строительства зданий и сооружений.

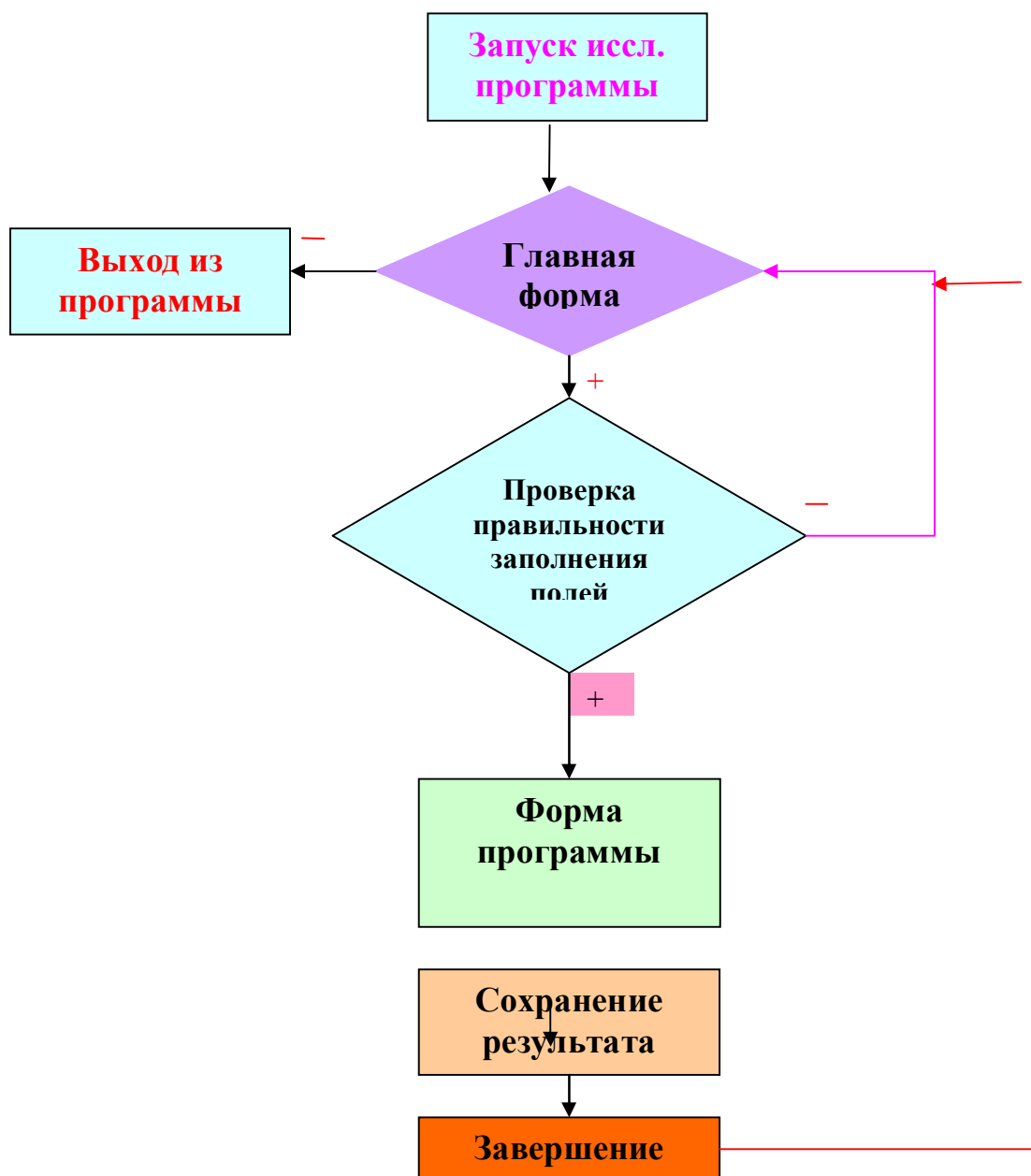


Рис. 1.1. Структурная схема реализации алгоритма исследовательской программы

Во время выполнения исследовательской работы было разработано программное средство, вычисляющее значение функции по заданному значению аргумента, выполняющегося с помощью средств вычислительной математики. В основе решения задачи лежит вычисление интерполяционным полиномом Лагранжа. Было создано четыре модуля, три из которых являются активными, четвертый («О программе») может быть

вызван с любого из трех главных модулей. Три основных модуля предлагают поэтапное выполнение задачи; в первом модуле вводятся начальные данные, во втором заполняется таблица со значениями функций и аргументов (это промежуточные точки, на основании которых и строится полином Лагранжа). Третий модуль является основным, использующим данные из второго и третьего модуля для вычислительного процесса. Интерфейс программы совместим с Windows-интерфейсом благодаря тому, что программа, в которой разрабатывалась исследовательская работа, – Delphi7 – предоставляет такую возможность и поддерживает именно Windows-интерфейс.

Список литературы

1. Кутуев М.Д., Укуев Б.Т., Матозимов Б.С., Мамбетов Э.М. Теория и практика сейсмозащиты сооружений /КГУСТА. – Б., 2010. – 372 с.
2. Матозимов Б.С., Токтонасаров Ж. М. Заявка №20060008.2, положительное решение о выдаче патента от 14.11.2006. Гибкое сейсмостойкое здание. – Бишкек, 2006. – 2 с.
3. Матозимов Б.С., Токтонасаров Ж.М. Анализ уравнений свободного незатухающего колебания // Известия ОшГУ. – Вып.2. – 2005. – С. 57-59.
4. Кутуев М.Д., Матозимов Б.С., Токтонасаров Ж.М. Экспериментальный проект гибкого здания с шарнирами пластичности //Механика и моделирование процессов технологии. Вып. 2. Казахстан, Тараз, 2005. – С.231-236.
5. Токтонасаров Ж. М., Матозимов Б.С. Стохастический метод расчета гибких зданий, жестко заделанных на фундамент, на сейсмостойкость //Вестник КГУСТА. – 2005. –Вып.2(8). – С.25-31.
6. Поляков С.В., Килимник Л.Ш., Черкашин А.В. Современные методы сейсмозащиты зданий. – М.: Стройиздат, 1989. – 320 с.
7. Хачиян Э.Е, Амбарцумян В.А. Динамические модели сооружений в теории сейсмостойкости. – М.: Наука, 1981. – 203 с.
8. Патент №92, заявка №20070014.2, рег. от 27.02.2009, Кыргызская Республика, KG 92 31.03.2009 «Сейсмостойкий дом» Ж.Ы.Маматов, Б.С.Матозимов, А.Ж.Андашев, Ж.Токтонасаров