

СОДЕРЖАНИЕ

Введение

1. Общая часть

- 1.1. Муниципальная информационная библиотечная система
- 1.2. Средства разработки

2. Специальная часть

- 2.1. Постановка задачи
- 2.2. Состав и структура библиографической записи
- 2.3. Описание задания
- 2.4. Описание алгоритма
- 2.5. Описание программы
- 2.6. Графическая схема алгоритма
- 2.7. Основные характеристики «ИРБИС»
- 2.8. Описание используемого языка
- 2.9. Техническое и программное обеспечение
- 2.10. Руководство по работе с БД «Читатель» в МИБС

3. Экономическая часть

- 3.1. Расчет затрат на разработку программы и решения задачи на ЭВМ
 - 3.1.1. Основная заработная плата
 - 3.1.2. Дополнительная заработная плата
 - 3.1.3. Отчисления в социальные статьи
 - 3.1.4. Стоимость работ на ВЦ
 - 3.1.5. Расчет стоимости материалов
 - 3.1.6. Косвенные расходы отдела по разработке программы
- 3.2. Расчет годовых затрат на эксплуатацию программы
- 3.3. Расчет годовых затрат на выполнение работ ранее употреблявшимся способом
- 3.4. Экономический анализ программы
- 3.5. Определение экономического эффекта программы
 - 3.5.1. Определение расчетного коэффициента экономической эффективности программы
- 3.6. Срок окупаемости разработанной программы
- 3.7. Календарный график работы

Заключение

Список используемой литературы

Приложение 1. Листинг программы

Приложение 2. Результат

					<i>ДП.2203.412.15.04.01</i>			
Изм.	Лист	№ докум.	Подпись	Дата				
Разраб.		Осташенко А.В.			Пояснительная записка	Лит.	Лист	Листов
Провер.		Карауш А.С.						
Рецензент		Гартман Е.В.						
н. контр.								
Утв.								
						ТПТ Гр. 412		

возможны только односторонние связи от старших вершин к младшим. Это облегчает доступ к необходимой информации, но если только все возможные запросы отражены в структуре дерева. Никакие иные запросы удовлетворены быть не могут.

Указанный недостаток снят в сетевой модели, где, по крайней мере теоретически, возможны связи «всех со всеми». Поскольку на практике это, естественно, невозможно, приходится прибегать к некоторым ограничениям. Использование иерархической и сетевой моделей ускоряет доступ к информации в базе данных. Но поскольку каждый элемент данных должен содержать ссылки на некоторые другие элементы, требуются значительные ресурсы, как дисковой, так и основной памяти. Недостаток основной памяти, конечно, снижает скорость обработки данных. Кроме того, для таких моделей характерна сложность реализации СУБД.

Объектно – ориентированная модель объединяет в себе две модели данных, реляционную и сетевую, и используется для создания крупных баз данных со сложными структурами данных.

					ДП.2203.412.15.04.01	Лист
Изм.	Лист	№ докум.	Подпись	Дата		

Муниципальная информационная библиотечная система взаимодействует с вузовскими и другими библиотеками по вопросам комплектования, повышения квалификации кадров, межбиблиотечного абонеента, а также принимает участие в подготовке и реализации совместных проектов. Сотрудники муниципальных библиотек информируют своих пользователей об информационных возможностях этих библиотек.

Все документы, поступающие в фонды библиотек, отражаются в электронном каталоге МИБС.

Корпоративная компьютерная сеть МИБС построена на основе Intranet/Internet технологиях и объединяет 10 библиотек.

Сеть МИБС состоит из 2 серверов и 46 компьютеров (из них 44 – с доступом к сети Интернет), в том числе 15 мест открытого доступа к собственным и удаленным ресурсам для пользователей библиотек.

Программа автоматизации МИБС основана на использовании программного обеспечения «ИРБИС». Освоены подсистемы программы, отвечающие за процессы комплектования и обработки документов, за регистрацию читателей.

1.2. Средства разработки

Разработки проводились на компьютере с процессором Intel Celeron 1.7 GHz, ОЗУ 512 Мб (DDR).

Программа написана в среде «ИРБИС» на языке управления библиографическими данными ISIS Pascal (CDS/ISIS).

Отчет оформлялся в Word 2002.

Распечатка проводилась на принтере Canon LBP-800.

					ДП.2203.412.15.04.01	Лист
Изм.	Лист	№ докум.	Подпись	Дата		

Файл **!21.PFT** предназначен для проверки данных поля «Год рождения» на недопустимые символы, а также для ограничения на дату рождения и проверки на дублетность.

Файл **!23.PFT** предназначен для создания переключателя для поля «Пол». Для этого создаем условие проверяющее наличие правильной буквы (М(м) -мужской или Ж(ж) - женский) в данном поле. Если же в данном поле вписана не верная буква то на экран выводится сообщение об ошибке.

Файл **!30.PFT** предназначен для создания ограничения на длину данных в поле «Единый идентификатор читателя» и проверки на дублетность.

Файл **!204.PFT** предназначен для проверки на наличие данных в подполях «Дата регистрации в читальном зале» и «Дата регистрации в Компьютерном зале», а также проверки регистрационного номера на дублетность и на его ввод в правильном формате.

Файл **!dbnflc.PFT** предназначен для проверки состояния поля «Прописка» (временная она или постоянная; если временная, то до какого года, т.е. не истекла ли она), а также проверки подполей «Регистрационный номер для чит. зала» и «Регистрационный номер для комп. зала» поля «Перерегистрация» на дублетность и на правильный формат ввода.

Файл **autoin.gbl** предназначен для добавления подполей, полей и изменении данных при перерегистрации читателей.

Файл **RDR.WS** предназначен для создания РЛ подполя поля «Перерегистрация».

Файл **!2000.WSS** предназначен для создания подполей в поле «Перерегистрация».

Файл **12.MNU** предназначен для создания переключателей да/нет.

Файл **RDRCW.FST** предназначен для создания префиксов для терминов словаря.

					ДП.2203.412.15.04.01	Лист
Изм.	Лист	№ докум.	Подпись	Дата		

2.7. Основные характеристики «ИРБИС»

Система автоматизации библиотек «ИРБИС» ориентирована на работу в локальных вычислительных сетях любого типа без ограничения количества пользователей при условии, что клиентской платформой является Windows 95/98/2000/NT и обеспечивается доступ к файл-серверу.

Система автоматизации библиотек «ИРБИС» представляет собой типовое интегрированное решение для автоматизации библиотечных технологий и предназначена для использования в условиях библиотек любого типа и профиля.

Система позволяет создавать и поддерживать любое количество баз данных, составляющих Электронный каталог (ЭК) или представляющих собой проблемно-ориентированные библиографические базы данных (БД).

Система предлагает большой набор сервисных средств, обеспечивающих удобство и наглядность пользовательских интерфейсов, упрощающих процесс ввода, исключая ошибки и дублирование информации.

Система обладает широкими возможностями для адаптации ее к условиям работы конкретной библиотеки, т.е. при передаче и установке системы может производиться ее настройка в соответствии со специфическими требованиями пользователя - начиная со структуры библиографического описания и кончая пользовательскими режимами работы.

Система является в достаточной мере открытой, что позволяет пользователю самостоятельно вносить изменения в широких пределах: от

1- означает отрицательный результат и обнаруженные ошибки считаются непреодолимыми, т.е. подлежат обязательному устранению;

2 - означает отрицательный результат, но при этом ошибки считаются преодолимыми, т.е. их можно не исправлять.

Остальной результат форматирования (начиная со второго символа) в случае отрицательного контроля выдается пользователю в качестве сообщения.

CDS/ISIS имеет особенности, позволяющие ему быть системой коллективного доступа. Основной особенностью, предоставляющей такую возможность является блокировка записи и блокировка базы данных. Блокировка записи запрещает более чем одному пользователю войти в запись для ее редактирования. Если бы несколько пользователей могли бы одновременно редактировать запись, возник бы хаос, поскольку при редактировании происходит копирование записи на локальную рабочую станцию, и окончательной версией была бы версия, выполненная последней. Блокировка базы данных не дает пользователю доступа в базу данных во время выполнения определенных функций системы, например, инвертирования. Это необходимо, потому что, если бы запись редактировалась во время полного инвертирования, происходило бы искажение индексного файла, поскольку во время инвертирования каждая запись просматривается несколько раз. Блокировка записи применяется во время ввода записей, а блокировка базы данных используется, например, во время экспорта или импорта записей.

Язык пакетной корректировки.

При решении поставленной задачи при вводе данных выполняется проверка данных на корректность ввода и корректировка записей базы данных. Для корректировки записей используется язык пакетной (глобальной) корректировки.

Корректировка выполняется на уровне записи, полей и подполей. Можно задать следующие виды корректировки записи:

- изменить значение поля или подполя;

										Лист
										37
Изм.	Лист	№ докум.	Подпись	Дата	ДП.2203.412.15.04.01					

- заменить целиком поле или подполе;
- добавить в запись поле или подполе в поле;
- удалить поле или подполе;
- удалить запись;
- создать новую запись в указанной базе данных и перейти к ее наполнению.

Для решения данной задачи создается файл задания, в котором задаются операторы корректировки и в каждом операторе необходимые данные: обозначение поля или подполя, повторение поля, форматы, определяющие способ корректировки.

Создать файл задания можно в текстовом редакторе в соответствии со структурой файла. Файл задания представляет собой текстовый файл с расширением WS. Файл задания содержит последовательность операторов корректировки, каждый из которых состоит из нескольких строк. Операторы выполняются в порядке их следования, причем каждый оператор использует значения полей и/или подполей измененных, возможно, предыдущими операторами.

Операторы пакетной корректировки:

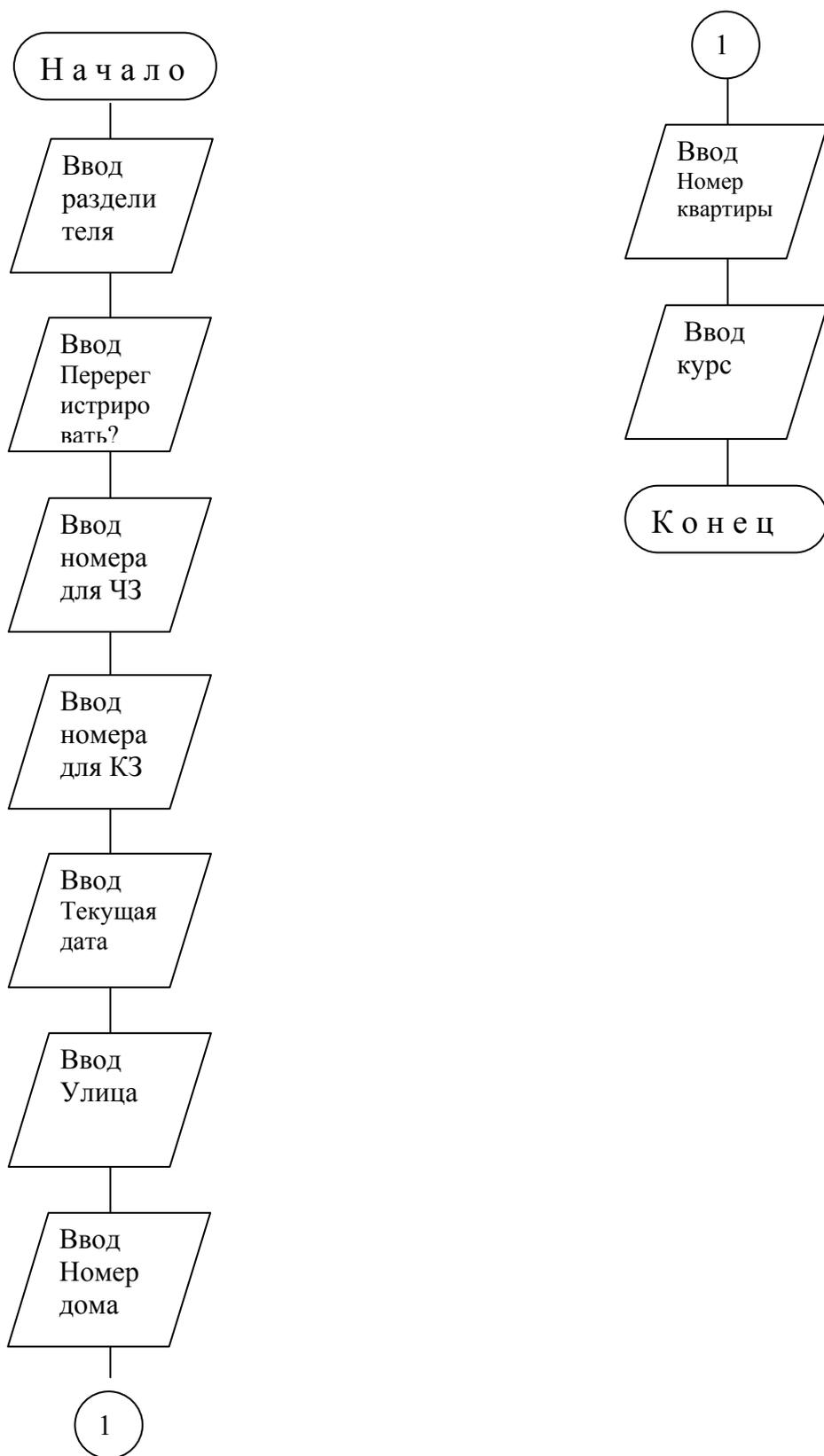
<i>ADD</i>	Выполняет добавление нового повторения поля или подполя в заданное существующее поле.
<i>REP</i>	Заменяет целиком поле или подполе на новое значение.
<i>CHA</i>	Выполняет замену данных в поле или в подполе. Позволяет приписывать данные в конец поля/подполя.
<i>DEL</i>	Удаляет поле или подполе.
<i>DELR</i>	Удаляет записи, поданные на корректировку. Не совместим ни с каким другим оператором корректировки.
<i>CORREC</i>	Определяет БД (текущую или другую), номера записей (MFN) в ней и дополнительные данные для проведения в этих записях пакетной корректировки. Сама корректировка выполняется операторами, следующими за CORREC и END.

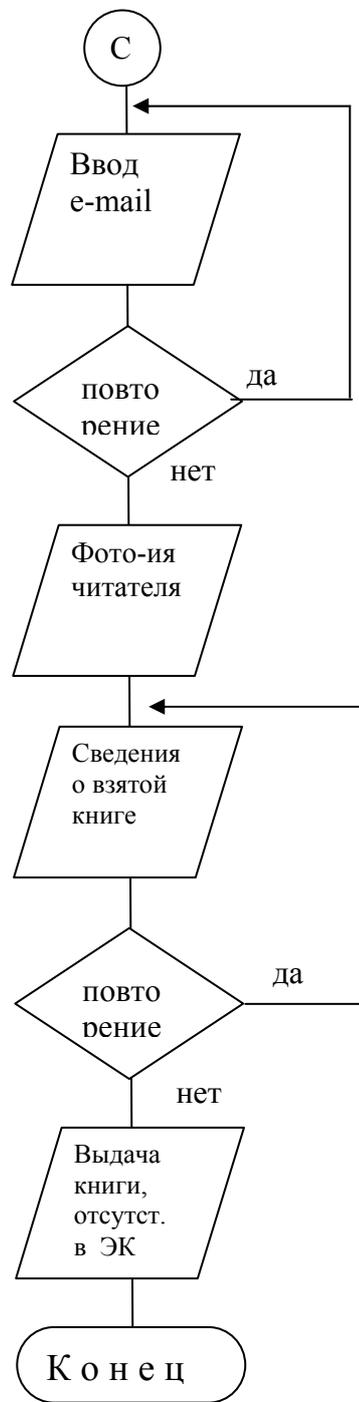
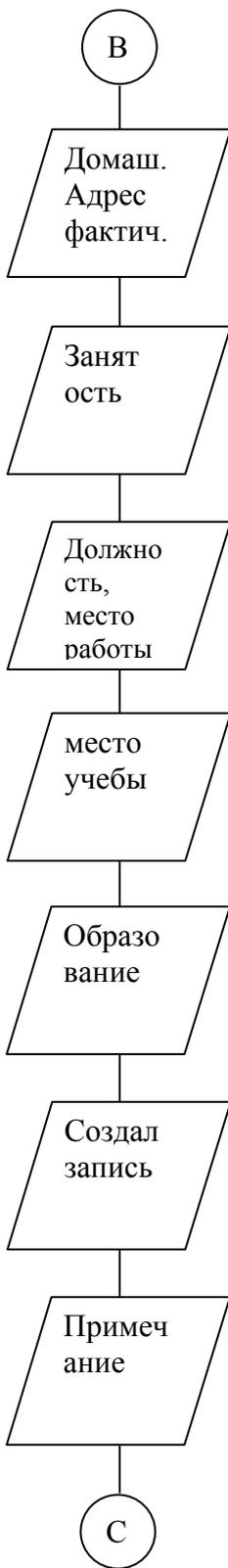
9. В верхней части окна (слева) выбираем подплоскость «Результаты поиска»;
10. На вкладке «Учетная карточка читателя» заполняем поле «Перерегистрация». Раскрываем подполе ввода (кнопка с многоточием в конце строки): в подполе «Перерегистрировать?» выбираем с помощью переключателя «Да»(1) и заполняем подполя «Номер для читального зала» и «Номер для компьютерного зала», заносим текущую дату одновременным нажатием клавиш «ALT» и «Д» в соответствующее подполе. Затем заполняются остальные подполя, при необходимости. После ввода всех данных нажимаем кнопку «Ввод». По окончании нажимаем кнопку «Сохранить» в верхней части окна.

					ДП.2203.412.15.04.01	Лист
Изм.	Лист	№ докум.	Подпись	Дата		

2.6. Графическая схема алгоритма

Файл 2000.WSS



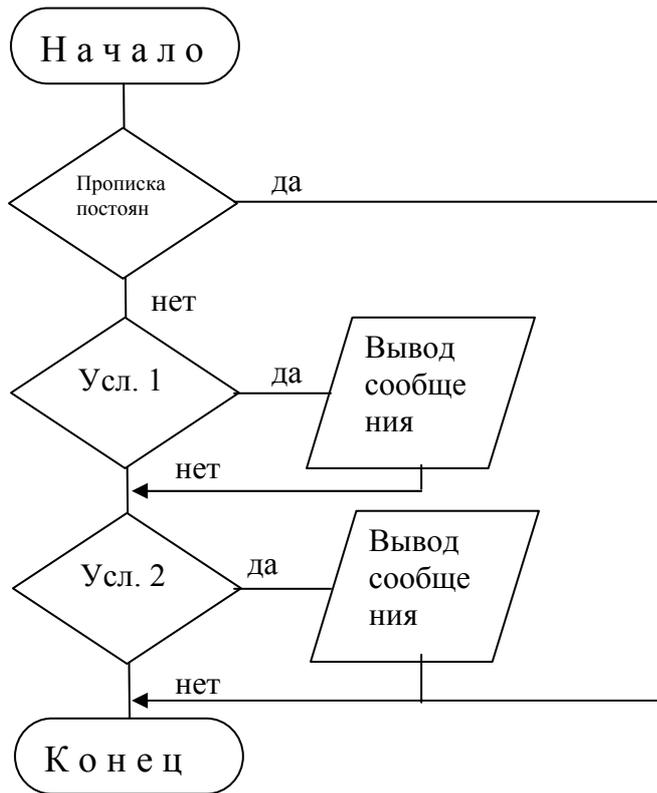


Изм.	Лист	№ докум.	Подпись	Дата

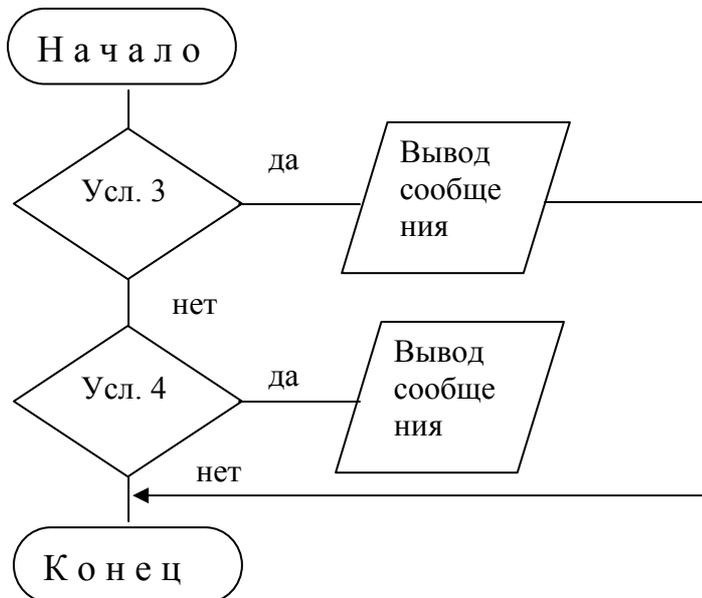
ДП.2203.412.15.04.01

Лист

Файл DBNFLC.PFT (часть)



Файл !13.PFT

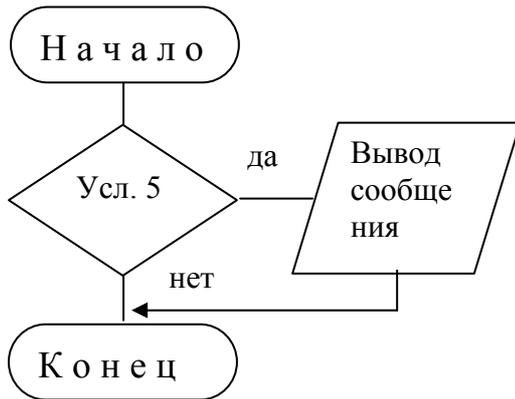


Изм.	Лист	№ докум.	Подпись	Дата

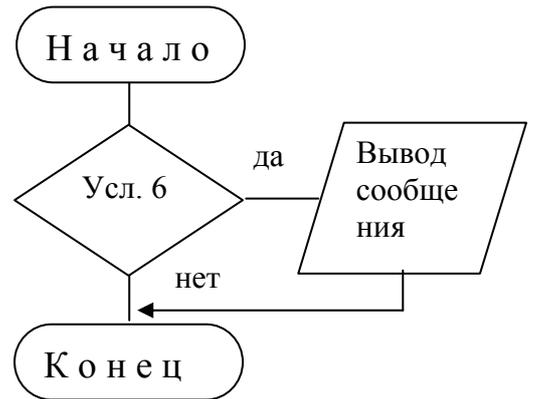
ДП.2203.412.15.04.01

Лист

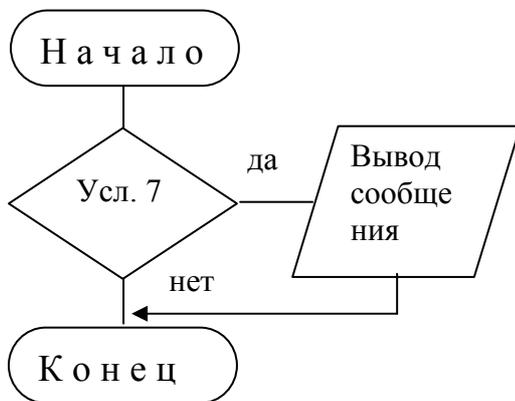
Файл !10. PFT



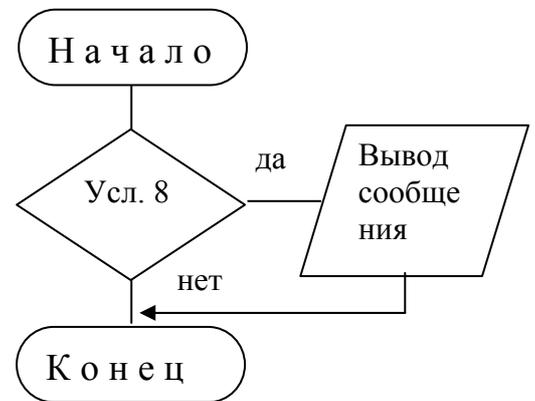
Файл !11. PFT



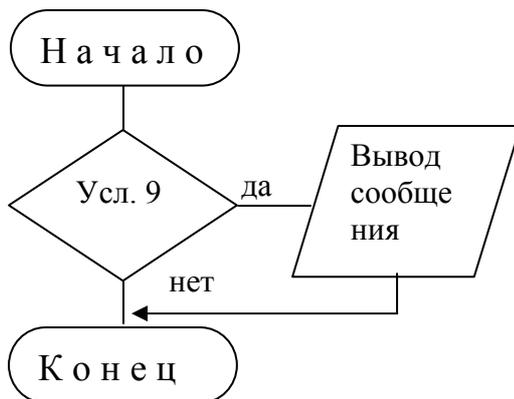
Файл !12. PFT



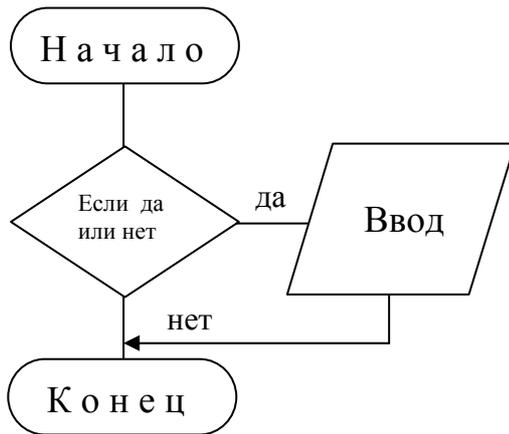
Файл !17. PFT



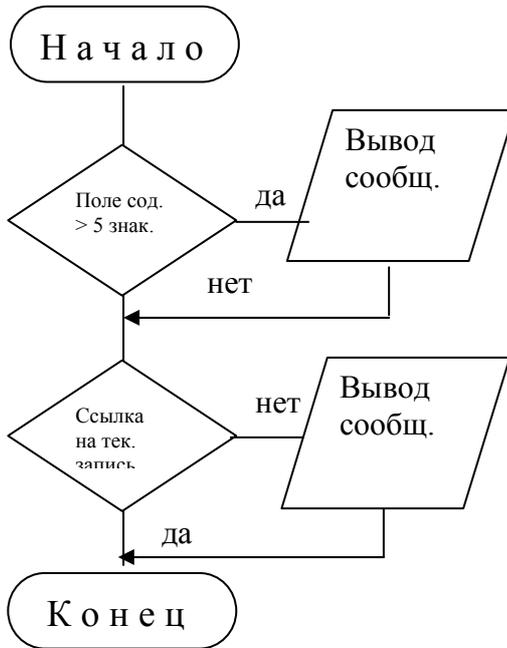
Файл !18. PFT



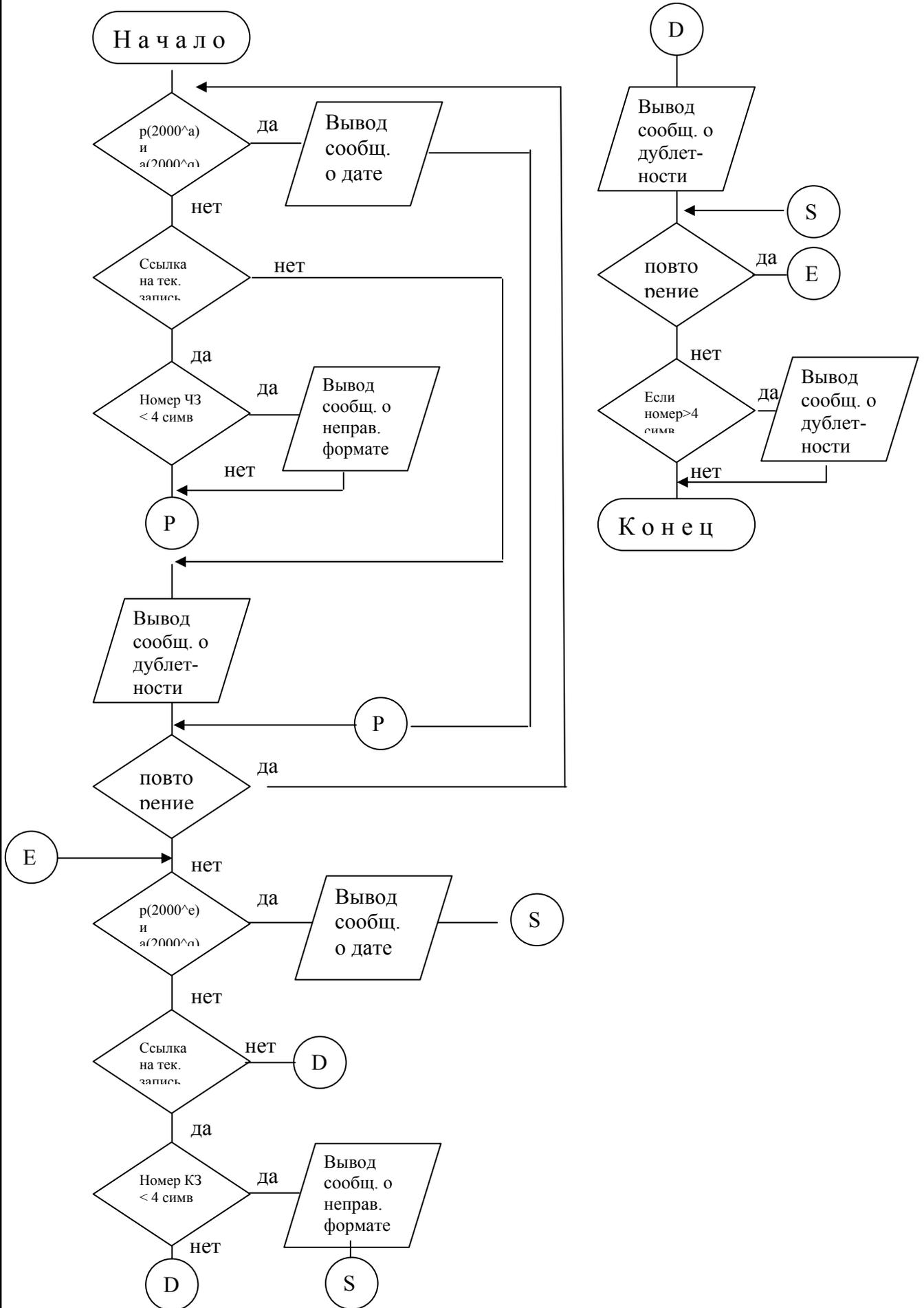
Файл 12. MNU



Файл !30.PFT



Файл !2000.PPT



Изм.	Лист	№ докум.	Подпись	Дата

ДП.2203.412.15.04.01



Комментарий к алгоритму программы:

- Усл. 1 – прописка временная и дата не указана
- Усл. 2 – прописка временная и тек. год больше даты
- Усл. 3 – Поле Улица существует, а поле номер дома не существует
- Усл. 4 – Поле Номер дома существует, а поле Улица не существует
- Усл. 5 – Поле Фамилия содержит цифры, символы или пробелы
- Усл. 6 – Поле Имя содержит цифры, символы или пробелы
- Усл. 7 – Поле Отчество содержит цифры, символы или пробелы
- Усл. 8 – Поле Телефон дом. содержит пробел или тире
- Усл. 9 – Поле Телефон служеб. содержит пробел или тире
- Усл. 10 – поле 204^а существует и поле 204^у пустое
- Усл. 11 – поле 204^е существует и поле 204^z пустое
- Усл. 12 – поле 2000^а существует и поле 2000^е нет
- Усл. 13 – поле 2000^е существует и поле 2000^а нет
- Усл. 14 – поле 2000^а существует и поле 2000^е нет
- Усл. 15 – поле 2000^е существует и поле 2000^а нет

					ДП.2203.412.15.04.01	Лист
Изм.	Лист	№ докум.	Подпись	Дата		

3.4. Экономический анализ программы

Рациональность разработки и внедрения той или иной задачи устанавливается на основе комплексного анализа целесообразности ее применения, которая завершается экономическим анализом, состоящим в установлении преимуществ и недостатков соответствующих средств и способов с точки зрения экономии общественно труда.

Современный этап развития науки и техники характеризуется многократностью решения одной и той же производственной задачи. Это в свою очередь предлагает проведение реальных расчетов сравнительной экономической эффективности вариантов. Экономический анализ сводится к определению по сравнительным вариантам экономического эффекта, получаемого предприятием.

					ДП.2203.412.15.04.01	Лист
Изм.	Лист	№ докум.	Подпись	Дата		

3.5. Определение экономического эффекта программы

При расчете экономического эффекта может быть получен отрицательный результат. Это говорит о том, что выбранный метод с точки зрения экономического анализа менее эффективен.

Годовая экономия определяется по формуле:

$$\text{Э}_{\text{год.}} = \text{С}_{\text{р.сп.год.}} - \text{С}_{\text{р.м.год.}},$$

где $\text{С}_{\text{р.сп.год.}}$ – годовые затраты на решение ранее применявшимся способом,

$\text{С}_{\text{р.м.год.}}$ – годовые затраты машинного решения задачи

$$\text{Э}_{\text{год.}} = 20496,63 - 12014,58 = 8482,05 \text{ руб.}$$

3.5.1. Определение расчетного коэффициента экономической эффективности программы

Расчет экономической эффективности проводится для определения наиболее экономически выгодного варианта обработки информации. Экономическая эффективность характеризует отношение общей величины эффекта к затратам, его вызвавшим.

$$\text{Е}_{\text{р.}} = \text{Э}_{\text{год.}} / (\text{С} + \text{С}_{\text{р.м.год.}}),$$

где С – себестоимость программы,

$\text{С}_{\text{р.м.год.}}$ – годовые затраты машинного решения задачи

$$\text{Е}_{\text{р.}} = 8482,05 / (2782 + 12014,58) = 0,6$$

										Лист
Изм.	Лист	№ докум.	Подпись	Дата	ДП.2203.412.15.04.01					

3.6. Срок окупаемости разработанной программы

Срок окупаемости характеризуют период времени, в течении которого общие затраты на составление программы возмещаются за счет экономии текущих расходов. Срок окупаемости представляет собой отношение затрат на составление программы к экономическому эффекту от его внедрения.

$$T_o = (C + C_{p.m.год.}) / \mathcal{E}_{год.} = 1 / E_p ,$$

где T_o – срок окупаемости

$$T_o = 1 / 0,6 = 1,6 \text{ года}$$

Из расчетов вытекает, что программа окупится за 19 месяцев.

Выбранный способ решения считается экономически эффективным.

					ДП.2203.412.15.04.01	Лист
Изм.	Лист	№ докум.	Подпись	Дата		

Листинг программы

Файл !10.PFT

```

/* предназначен для проверки поля «Фамилия» на наличие недопустимых
/* символов
/* 1 результата форматирования означает отрицательный результат и
/* обнаруженные ошибки считаются непреодолимыми, т.е. подлежат
/* обязательному устранению

```

```

if v10:'0' or v10:'1' or v10:'2' or v10:'3' or v10:'4' or v10:'5' or v10:'6'
    or v10:'7' or v10:'8' or v10:'9' or v10:',' or v10:',' or v10:',' or v10:','
    or v10:'@' or v10:'№' or v10:'%' or v10:':' or v10:',' or v10:'$' or v10:','
    or v10:'(' or v10:')' or v10:'=' or v10:'+' or v10:'\' or v10:'|' or v10:' '
    then '1 Вы ввели недопустимый символ (цифру, пробел)!!!'
fi

```

Файл !11.PFT

```

/* предназначен для проверки поля «Имя» на наличие недопустимых
/* символов

```

```

if v11:'0' or v11:'1' or v11:'2' or v11:'3' or v11:'4' or v11:'5' or v11:'6'
    or v11:'7' or v11:'8' or v11:'9' or v11:',' or v11:',' or v11:',' or v11:','
    or v11:'!' or v11:'@' or v11:'№' or v11:'%' or v11:':' or v11:',' or v11:'$'
    or v11:',' or v11:'(' or v11:')' or v11:'=' or v11:'+' or v11:'\' or v11:'|'
    then '1 Цифры или символы или пробелы не допустимы в поле Имя!!!'
fi/

```

Файл !12.PFT

```

/* предназначен для проверки поля «Отчество» на наличие недопустимых
/* символов

```

```

if v12:'0' or v12:'1' or v12:'2' or v12:'3' or v12:'4' or v12:'5' or v12:'6'
    or v12:'7' or v12:'8' or v12:'9' or v12:',' or v12:',' or v12:',' or v12:','
    or v12:'!' or v12:'@' or v12:'№' or v12:'%' or v12:':' or v12:',' or v12:'$'
    or v12:',' or v12:'(' or v12:')' or v12:'=' or v12:'+' or v12:'\' or v12:'|'
    then '1 Цифры или символы или пробелы не допустимы в поле Отчество!!!'
fi/

```

Файл !13.PFT

```

if p(v13^d) and a(v13^e)
    then /1 Не указан номер дома! Введите обязательно!
    else
        if p(v13^e) and a(v13^d)

```

```
then /'1 Не указана улица! Введите обязательно!
```

```
fi
```

```
fi
```

Файл !17.PFT

```
if v17:'' or v17:'-'
```

```
then '1 Номер телефона пишутся слитно!!!'
```

```
fi
```

Файл !18.PFT

```
if v18:'' or v18:'-'
```

```
then '1 Номер телефона пишется слитно!!!'
```

```
fi
```

Файл !2000.PFT

```
/* предназначен для проверки подполей вложенного РЛ поля
```

```
/* «Перерегистрация» на наличие данных в подполе «Текущая дата»,
```

```
/* проверки регистрационного номера на дублетность и на правильность
```

```
/* формата
```

```
/* проверка подполей для читального зала
```

```
(if p(v2000^a) and a(v2000^q)
```

```
/* проверка на ввод даты
```

```
then '1 Не введена текущая дата!'
```

```
else
```

```
if L('IDLR='v2000^a,'-',v2000^q*0.4)<=0 or val(ref(L("IDLR="v2000^a,'-',v2000^q*0.4),mfn))=val(mfn)
```

```
then
```

```
if (p(v2000^a) and val(v2000^a)<1000 and ((val(v2000^a)<10 and v2000^a*0.3<>'000')
```

```
or (val(v2000^a)<100 and v2000^a*0.2<>'00') or (val(v2000^a)<1000 and v2000^a*0.1<>'0'))
```

```
then /'1 Неправильный формат регистрационного номера в ЧЗ!'
```

```
else /'0'
```

```
fi
```

```
else /'1 Дублетный регистрационный номер читателя ЧЗ  
'v2000^a,ref(L('IDLR='v2000^a,'-',v2000^q*0.4), ' (см. N ' f(val(mfn),0,0),'))'
```

```
fi
```

```
fi /)
```

```
/* проверка подполей для компьютерного зала
```

```
(if p(v2000^e) and a(v2000^q)
```

```
then '1 Не введена текущая дата!'
```

```

else
  if L('IDLK='v2000^e,'-',v2000^q*0.4)<=0 or val(ref(L("IDLK="v2000^e,'-',v2000^q*0.4),mfn))=val(mfn)
  then
    if (p(v2000^e) and val(v2000^e)<1000 and ((val(v2000^e)<10 and v2000^e*0.3<>'000')
      or (val(v2000^e)<100 and v2000^e*0.2<>'00') or (val(v2000^e)<1000 and v2000^e*0.1<>'0'))))
    then '1 Неправильный формат регистрационного номера в КЗ!'
    else /0'
  fi
  else /'1 Дублетный регистрационный номер читателя в КЗ
  ',v2000^e,ref(L('IDLK=',v2000^e,'-',v2000^q*0.4), '(см. N ' f(val(mfn),0,0),'))'
  fi
fi/)
if v2000^a*4<>" or v2000^e*4<>"
  then /'1 Регистрационный номер должен быть четырехзначным!'
fi

```

Файл !204.PFT

```

/* предназначен для проверки поля Регистрация МБ «Компьютерный мир» на
/* наличие данных в подполе «Дата регистрации», проверки
/* регистрационного номера на дублетность и на правильность формата

```

```

/* проверка подполей для читального зала

```

```

(if p(v204^a) and a(v204^y) /* проверка на ввод даты
  then '1 Не введена Дата регистрации в ЧЗ библиотеки =Комп.мир='
  else
    if L('IDLR='v204^a,'-',v204^y*0.4)<=0 or val(ref(L("IDLR="v204^a,'-',v204^y*0.4),mfn))=val(mfn)
    then
      if (p(v204^a) and val(v204^a)<1000 and ((val(v204^a)<10 and v204^a*0.3<>'000')
        or (val(v204^a)<100 and v204^a*0.2<>'00') or (val(v204^a)<1000 and v204^a*0.1<>'0'))))
      then /'1 Неправильный формат регистрационного номера в ЧЗ!'
      else /0'
    fi
    else /'2 Дублетный регистрационный номер читателя ЧЗ
    ',v204^a,ref(L('IDLR=',v204^a,'-',v204^y*0.4),
      '(см. N ' f(val(mfn),0,0),'))'
    fi
fi /)

```

/* проверка подполей для компьютерного зала

```
(if p(v204^e) and a(v204^z)
  then '1 Не введена Дата регистрации в КЗ библиотеки =Комп.мир='
  else
    if L('IDLK='v204^e,'-',v204^z*0.4)<=0 or val(ref(L("IDLK="v204^e,'-',v204^z*0.4),mfn))=val(mfn)
      then
        if (p(v204^e) and val(v204^e)<1000 and ((val(v204^e)<10 and v204^e*0.3<>'000')
          or (val(v204^e)<100 and v204^e*0.2<>'00') or (val(v204^e)<1000 and v204^e*0.1<>'0'))
            then /*1 Неправильный формат регистрационного номера в КЗ!
              else /*0
                fi
            else /*2 Дублетный регистрационный номер читателя в КЗ
              'v204^e,ref(L('IDLK='v204^e,'-',v204^z*0.4),
                ' (см. N ' f(val(mfn),0,0),'))
            fi
          fi /)
```

```
if v204^a*4<>" or v204^e*4<>"
  then /*1 Регистрационный номер должен быть четырехзначным!
  fi
```

Файл !30.PFT

/* предназначен для проверки поля «Идентификатор» на длину и на
/* дублетность

/* проверка на длину

```
if v30*5<>"
  then /*1 Идентификатор должен быть пятизначный!!!
  fi,
  /* проверка на дублетность
  if L("ID="v30)<=0 or val(ref(l("ID="v30),mfn))=val(mfn)
    then /*0
    else /*1 Дублетный идентификатор читателя: ', v30,
      ref(L("ID="v30),'(см.№',f(val(mfn),0,0),'))
    fi
```

Файл 12.MNU

/* предназначен для создания переключателей да/нет для поля
/* «Перерегистрация»

1
да
0
нет

Файл 2000.WSS

/* предназначен для создания вложенного рабочего листа для поля
/* «Перерегистрация» и ввода его элементов

8
1
<1> Перерегистрировать?
0
0
7
12.mnu

A
<\$A> Номер для чит. зала
0
0
2
,NOM2041=,

Регистрационный номер должен быть четырехзначным. При необходимости добавьте впереди ноль.

E
<\$E> Номер для комп.зала
0
0
2
,NOM2045=,

Регистрационный номер должен быть четырехзначным. При необходимости добавьте впереди ноль.

Q

<\$Q> Текущая дата

0

0

0

Для ввода даты нажмите <Alt> + <D>

D

Адрес: улица

0

0

2

,UL=,

K

<\$K> Адрес: номер дома

0

0

0

F

<\$F> Адрес: номер квартиры

0

0

0

U
Курс
0
0
0

Файл AUTOIN.GBL

/* предназначен для глобальной корректировки поля Регистрация МБ
/* «Компьютерный мир» и «Перерегистрация»

```
0
IF
v2000^1 /* проверка на наличие подполя ^1 в поле 2000
XXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXX
IF
if p(v2000^a) and a(v2000^e) then if v204^y:v2000^q.4 then '0' else if
v204^z:v2000^q.4 then '1' else '0' fi fi else if a(v2000^a) and p(v2000^e)
then if v204^z:v2000^q.4 then '0' else if v204^y:v2000^q.4 then '1' else '0' fi fi fi fi
XXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXX
ADD
2001 /* добавление поля 2001
XXXXXXXXXXXXXXXXXXXXXXXXX
'1'
XXXXXXXXXXXXXXXXXXXXXXXXX
FI
XXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXX
IF
if p(v2001) then '1' else '0' fi /* если поле 2001 существует
```

XXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXX

ADD /* добавляем подполе в поле 204
204^v

L
if p(v2000^a) and a(v2000^e) then '^A'v2000^a'^Y'v2000^q else
if a(v2000^a) and p(v2000^e) then '^E'v2000^e'^Z'v2000^q fi fi

XXXXXXXXXXXXXXXXXXXXX

DEL /* удаление подполя
204^v

L /* последнее повторение

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

FI

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

IF

if a(v2001) then '1' else '0' fi /* если поле не существует

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

ADD /* добавления поля

204

XXXXXXXXXXXXXXXXXXXXX

if p(v2000^a) and a(v2000^e) then '^A'v2000^a'^Y'v2000^q else if a(v2000^a) and
p(v2000^e) then '^E'v2000^e'^Z'v2000^q else if p(v2000^a) and p(v2000^e) then
'^A'v2000^a'^Y'v2000^q'^E'v2000^e'^Z'v2000^q fi fi fi

XXXXXXXXXXXXXXXXXXXXX

FI

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

FI

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

IF

if v2000^1='1' then '1' else '0' fi

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

REP /* замена подполя «Курс» на новое значение

19^D

*

if p(v2000^U) then v2000^U else if a(v2000^u) then v19^D fi fi

XXXXXXXXXXXXXXXXXXXXX

REP /* замена подполя «Улица» на новое значение

13^D

if p(v2000^D) then v2000^D else if a(v2000^D) then v13^D fi fi

XXXXXXXXXXXXXXXXXXXXX

REP /* замена подполя «Номер дома» на новое

13^E /* значение

if p(v2000^K) then v2000^K else if a(v2000^K) then v13^E fi fi

XXXXXXXXXXXXXXXXXXXXX

REP /* замена подполя «Номер квартиры» на новое

13^F /* значение

if p(v2000^F) then v2000^F else if a(v2000^F) then v13^F fi fi

XXXXXXXXXXXXXXXXXXXXX

FI

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

DEL /* удаление содержимого поля

2000

*

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

ADD /* добавление поля с новым значением

2000

XXXXXXXXXXXXXXXXXXXXX

'^10'

XXXXXXXXXXXXXXXXXXXXX

DEL /* удаление поля

2001

*

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXX

Файл DBNFLC.PFT

```
/* Данный файл предназначен для формально-логического контроля при
/* сохранении документа, в т.ч. и для проверки состояния поля «Прописка»;
/* 2 результата форматирования означает отрицательный результат, но при
/* этом ошибки считаются непреодолимыми, т.е. их можно исправлять
```

```
if &unifor("E3"v13^g)='постоянная'
  then /'0' /* правильно
  else
    if v13^g:'временная' and val(v13^g)=0 or (val(v13^g)=0)
      then /'2 Не указана дата прописки'
      else
        if &unifor("E3"v13^g):'временная' and &unifor('3')>&unifor('F2'v13^g)
          then /'2 Срок прописки истек'
        fi
      fi
    fi
fi/
```

Файл РАБОТА.MNU

```
/* предназначен для создания переключателей в поле «Занятость»
```

дошкольник

1-дошк.

школьник

2-школ.

студент

3-студ.

аспирант

4-асп.

работающий

5-работ.

безработный

6-нераб.

пенсионер

7-пенс.

Файл RDR.WS

/* предназначен для заполнения рабочих листов (РЛ) в рабочей области Ввод
 /* Каждая строка на странице РЛ представляет собой описание одного
 /* элемента ввода.
 /* **Метка поля** – числовая метка поля.
 /* Вводим произвольно или выбираем из списка;
 /* **Наименование** – наименование поля.
 /* Вводим произвольно или выбираем из списка;
 /* **Повторение** – повторяемость поля. Выбираем из предлагаемого списка
 /* (1-повторяющееся, 0-неповторяющееся);
 /* **[резерв1]** – в настоящее время не используется;
 /* **Режим** – использование расширенных средств ввода, т.е. Доп. Инф.:
 /* **0** – в этом случае параметр Доп. Инф. не используется;
 /* **1**- ввод через меню (справочник);
 /* **2**- ввод через словарь;
 /* **5** – ввод через вложенный РЛ (РЛ подполей), т.е. ввод поля с подполями;
 /* **7** – ввод с использованием переключателей;

4	/* количество рабочих листов
1 Учетная карточка читателя	/* наименование рабочих листов
2 Регистрация МИБС	
3 Дополнительные данные о читателе	
4 Сведения о взятой литературе	
10	/* разделители подполей
20	
11	
2	
2000	/* метка поля
Перерегистрация	/* наименование поля
0	/* повторяемость поля
0	
5	/* ввод через вложенный РЛ
2000.wss	/* имя файла РЛ подполей
@!2000	
Заполните поля для перерегистрации	/* подсказка
^10	
10	
<10> Фамилия	
0	
0	
0	
@!10	
Данное поле не должно содержать символы и пробелы	

11

<11> Имя

0

0

0

@!11

Данное поле не должно содержать символы и пробелы

12

<12> Отчество

0

0

0

@!12

Данное поле не должно содержать символы и пробелы

21

<21> Год рождения

0

2

/* ввод через словарь

0

@!21

30

<30> Единый идентификатор читателя

0

0

2

,ID=,

@!30

Идентификатор читателя должен быть пятизначным

23

<23> Пол

0

0

7
pol.mnu
@!23

/* ввод с использованием переключателей

17
<17> Телефон домашний
0
0
0

@!17
Номер телефона пишется слитно

18
<18> Телефон служебный
1
0
0

@!18
Номер телефона пишется слитно

51
<51> Дата записи
1
0
0

Для ввода текущей даты <Alt+Д>

204
<204> МБ "Компьютерный мир"
1
0
5
204.wss
@!204
Данные только для МБ "Компьютерный мир"

14

<14> Паспортные данные

0

0

5

14!.wss

Буквы серии паспорта заполняются только РУССКИМИ буквами !!!

13

<13> Домашний адрес (по прописке)

0

0

5

13.wss

@!13

Если адреса совпадают, то заполняется только адрес прописки

^СТомск

130

<130> Домашний адрес (фактический)

0

0

5

130.wss

@!130

15

<15> Занятость (категория)

0

0

7

rabota.mnu

/* ввод через меню справочник

16

<16> Должность, место работы

0

0

0

19

<19> Место учебы

0

0

5

19.wss

20

<20> Образование

0

0

7

rdro.mnu

31

<31> Создал запись (ФИО регистратора)

0

0

2

,FIO=,

/* префикс для терминов словаря

33

<33> Примечание

0

0

2

,PRIM=,

Инвалиды, ветераны и др. категории населения с льготами

32

<32> E-mail

1

0

0

950

<950> Фотография читателя (файл)

0

0

0

Указывать имя файла с расширением (без пути)

40

<40> Сведения о взятой книге

1

0

5

v40.wss

Заполняется автоматически программой ИРБИС

200

<200> Выдача книги, ОТСУТСТВУЮЩЕЙ в ЭК

0

0

5

200r.wss

@!200r

Файл RDRCW.FST (часть файла)

/* предназначен для создания префиксов для терминов словаря

2000 0 ("IDLR=",v2000^a,"-",v2000^q.4/)
2000 0 ("IDLR=",v2000^e,"-",v2000^q.4/)
204 0 ("IDLR=",v204^a,"-",v204^y.4/)
204 0 ("IDLK=",v204^e,"-",v204^z.4/)

Файл RDRO.MNU

/* предназначен для создания переключателей для поля «Образование»

начальное	/* значение
нач.	/* переключатель
неполное среднее	
неп.ср.	
среднее	
сред.	
среднее специальное	
ср.-спец.	
неполное высшее	
н.высш.	
высшее	
высш.	