МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ

ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет Информационных Технологий

Кафедра Программной инженерии

Специальность 6-05-0612-01 Программное обеспечение информационных технологий

Специализация Программирование интернет-приложений

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**К КУРСОВОМУ ПРОЕКТУ НА ТЕМУ:**

«Разработка компилятора BNA-2024»

Выполнил студент Борисов Никита Андреевич

(Ф.И.О.)

Руководитель проекта

(учен. степень, звание, должность, подпись, Ф.И.О.)

Заведующий кафедрой

(учен. степень, звание, должность, подпись, Ф.И.О.)

Консультанты

(учен. степень, звание, должность, подпись, Ф.И.О.)

Нормоконтролер

(учен. степень, звание, должность, подпись, Ф.И.О.)

Курсовой проект защищен с оценкой

Минск 2024

**Содержание**

[Введение 3](#_Toc181355806)

[Глава 1. Спецификация языка программирования 4](#_Toc181355807)

[1.1 Характеристика языка программирования 4](#_Toc181355808)

[1.2 Алфавит языка 4](#_Toc181355809)

[1.3 Символы сепараторы 5](#_Toc181355810)

[1.4 Применяемые кодировки 5](#_Toc181355811)

[1.5 Типы данных 5](#_Toc181355812)

[1.6 Преобразование типов данных 6](#_Toc181355813)

[1.7 Идентификаторы 6](#_Toc181355814)

[1.8 Литералы 6](#_Toc181355815)

[1.9 Область видимости идентификаторов 7](#_Toc181355816)

[1.10 Инициализация данных 7](#_Toc181355817)

[1.11 Инструкции языка 7](#_Toc181355818)

[1.12 Выражения и их вычисления 8](#_Toc181355819)

[1.13 Программные конструкции языка 8](#_Toc181355820)

[1.14 Область видимости 9](#_Toc181355821)

[1.15 Семантические проверки 9](#_Toc181355822)

[1.16 Распределение оперативной памяти на этапе выполнения 9](#_Toc181355823)

[1.17 Стандартная библиотека и её состав 9](#_Toc181355824)

[1.18 Ввод и вывод данных 10](#_Toc181355825)

[1.19 Точка входа 10](#_Toc181355826)

[1.20 Препроцессор 10](#_Toc181355827)

[1.21 Соглашения о вызовах 10](#_Toc181355828)

[1.22 Объектный код 10](#_Toc181355829)

[1.23 Классификация сообщений транслятора 11](#_Toc181355830)

Введение

Целью курсового проекта поставлена задача разработки компилятора для моего языка программирования BNA-2024. Этот язык программирования предназначен для выполнения простейших операций и арифметических действий над числами.

Компилятор BNA-2024 – это программа, задачей которого является перевод программы, написанной на языке программирования BNA-2024 в программу на язык ассемблера.

Транслятор BNA-2024 состоит из следующих частей:

– лексический и семантический анализаторы;

– синтаксический анализатор;

– генератор исходного кода на языке ассемблера.

Исходя из цели курсового проекта, были определены следующие задачи:

– разработка спецификации языка программирования;

– разработка структуры транслятора;

– разработка лексического и семантического анализаторов;

– разработка синтаксического анализатора;

– преобразование выражений;

– генерация кода на язык ассемблера;

– тестирование транслятора.

Решения каждой из поставленных задач будут приведены в соответствующих главах курсового проекта.

Глава 1. Спецификация языка программирования

## **Характеристика языка программирования**

Язык программирования BNA-2024 классифицируется как процедурный, универсальный, строготипизированный, компилируемый и не объектно-ориентированный язык.

* 1. **Алфавит языка**

Алфавит языка BNA-2024 основан на кодировке Windows-1251, представленной на рисунке 1.1.



Рисунок 1.1 – Алфавит входных символов

На этапе выполнения могут использоваться символы латинского алфавита, кириллица нижнего регистра, цифры десятичной системы счисления от 0 до 9, спецсимволы, а также непечатные символы пробела, табуляции и перевода строки.

## **Символы сепараторы**

Символы, которые являются сепараторами представлены в таблице 1.1.

|  |  |  |
| --- | --- | --- |
| Сепаратор | Название | Область применения |
| ‘ ‘ | Пробел | Допускается везде, кроме идентификаторов и ключевых слов |
| ; | Точка с запятой | Разделение конструкций |
| {…} | Фигурные скобки | Заключение программного блока |
| (…) | Круглые скобки | Приоритет операций, параметры функции |
| […] | Квадратные скобки | Заключение блоков if, else, while |
| ‘…’ | Одинарные кавычки | Допускается везде, кроме идентификаторов и ключевых слов |
| = | Знак «равно» | Присваивание значения |
| , | Запятая | Разделение параметров |
| ~  !  >  <  »  « | Знаки «равенство», «неравенство», «больше», «меньше», «больше или равно» , «меньше или равно» | Логические выражения |
| +  -  \*  /  %  @  $ | Знаки «сумма», «разность», «умножение», «деление», «деление по модулю», «сдвиг бита влево», «сдвиг бита вправо» | Арифметические выражения |

Таблица 1.1 – Сепараторы

## **Применяемые кодировки**

Для написания исходного кода на языке программирования BNA-2024 используется кодировка Windows-1251.

## **Типы данных**

В языке BNA-2024 реализованы два типа данных: целочисленный и строковый. Описание реализованных типов данных представлено в таблице 1.2.

|  |  |
| --- | --- |
| Тип данных | Описание типа данных |
| Беззнаковый целый uint | Фундаментальный тип данных. Используется для работы с целочисленными значениями. В памяти занимает 4 байта.  Максимальное значение: 4,294,967,295.  Минимальное значение: 0.  Инициализация по умолчанию: значение 0. |
| Строковый тип данных stringing | Фундаментальный тип данных. Используется для работы с символами, каждый символ в памяти занимает 1 байт.  Максимальное количество символов: 255.  Инициализация по умолчанию: длина 0, символ конца строки “\0”. |

Таблица 1.2 – Типы данных языка BNA-2024

## **Преобразование типов данных**

Преобразование типов данных не поддерживается, т.е. язык является строготипизированным.

## **Идентификаторы**

В имени идентификатора допускаются символы латинского алфавита и кириллица в нижнем. Максимальная длина имени идентификатора - 30 символов. Максимальная длина имени идентификатора функции - 30 символов. При вводе идентификатора длиной более разрешенного количества символов, он будет усекаться. Имя идентификатора не может совпадать с ключевыми словами и не может иметь имя, как функция, уже содержащаяся в стандартной библиотеке.

## **Литералы**

С помощью литералов осуществляется инициализация переменных. В языке существует два типа литералов. Краткое описание литералов языка BNA-2024 представлено в таблице 1.3.

|  |  |  |  |
| --- | --- | --- | --- |
| Тип литерала | Регулярное выражение | Описание | Пример |
| Целочисленный литерал | [1-9]+[0-9]\* | Целочисленные литералы, по умолчанию инициализируются 0. Литералы могут быть только rvalue. | declare uint sum;  sum = 15;  15 – целочисленный литерал. |
| Строковые литерал | [a-z|а-я|0-9|']+ | Символы, заключённые в '…' (одинарные кавычки), по умолчанию инициализируются пустой строкой. Литералы могут быть только rvalue. | declare string text = ‘text’;  text – строковый литерал. |

Таблица 1.3 – Описание литералов

## **Область видимости идентификаторов**

Область видимости «сверху вниз» (по принципу С++). В языке BNA-2024 требуется обязательное объявление переменной перед её инициализацией и последующим использованием. Все переменные должны находиться внутри программного блока. Имеется возможность объявления одинаковых переменных в разных блоках, т. к. переменные, объявленные в одной функции, недоступны в другой. Каждая переменная получает префикс – название функции, в которой она объявлена.

## **Инициализация данных**

При объявлении переменной не допускается инициализация данных. Краткое описание способов инициализации переменных языка BNA-2024 представлено в таблице 1.4.

|  |  |  |
| --- | --- | --- |
| Конструкция | Описание | Пример |
| declare <тип данных> <идентификатор>; | Автоматическая инициализация: переменные типа uint инициализируются нулём, переменные типа string – пустой строкой. | declare uint sum;  declare string text; |
| <идентификатор> = <значение>; | Присваивание переменной значения. | sum = 7;  string = ‘text’; |

Таблица 1.4 – Способы инициализации переменных

## **Инструкции языка**

Все возможные инструкции языка программирования BNA-2024 представлены в общем виде в таблице 1.5.

|  |  |
| --- | --- |
| Инструкция | Запись на языке BNA -2024 |
| Объявление переменной | declare <тип данных> <идентификатор>; |
| Присваивание | <идентификатор> = <значение>/<идентификатор>; |
| Объявление внешней функции | <тип данных> function <идентификатор> (<тип данных> <идентификатор>, …) |
| Блок инструкций | main  {  …  }; |
| Возврат из подпрограммы | return <идентификатор> / <литерал>; |
| Условная инструкция | if (условие) [/программный блок если условие верно/]  else [/программный блок если условие ложно/]; |
| Инструкция цикла | while(условие)[/программный блок если условие верно/]; |
| Вывод данных | output (<идентификатор> / <литерал>); |

Таблица 1.5 – Инструкции языка программирования BNA - 2024

**Операции языка**

Язык программирования BNA-2024 может выполнять операции сравнения, представленные в таблице 1.6.

|  |  |  |  |
| --- | --- | --- | --- |
| Операция | Примечание | Типы данных | Пример |
| = | Присваивание | (uint, uint)  (string, string) | sum = 7;  text = ‘text’; |
| < | Знак «меньше» для условной инструкции | (uint, uint) | if (sum < diff) […] |
| > | Знак «больше» для условной инструкции | (uint, uint) | if (sum > diff) […]  else […] |
| ( | Приоритет операций | - | sum = (a + b) - c; |
| ) |
| ~ | Равенство | (uint, uint) | a ~ b; |
| ! | Неравенство | (uint, uint) | a ! b; |
| « | Меньше или равно | (uint, uint) | a « b; |
| » | Больше или равно | (uint, uint) | a » b; |
| + | Сумма | (uint, uint) | a + b; |
| - | Разность | (uint, uint) | a - b; |
| \* | Умножение | (uint, uint) | a \* b; |
| / | Деление | (uint, uint) | a / b; |
| % | Деление по модулю | (uint, uint) | a % b; |
| @ | Сдвиг битов влево | (uint, uint) | a @ b; |
| $ | Сдвиг битов вправо | (uint, uint) | a $ b; |

Таблица 1.6 – Операции языка программирования BNA-2024

## **Выражения и их вычисления**

Круглые скобки в выражении используются для изменения приоритета операций. Выражение может содержать вызов функции, если эта функция уже содержится в стандартной библиотеке.

## **Программные конструкции языка**

Ключевые программные конструкции языка программирования BNA-2024 представлены в таблице 1.7.

|  |  |
| --- | --- |
| Конструкция | Запись на языке BNA-2024 |
| Главная функция (точка входа) | main  {  …  return <идентификатор> / <литерал>;  }; |
| Функция | function <тип> <идентификатор> (<тип> <идентификатор>, …)  {  …  return <идентификатор> / <литерал>;  }; |

Таблица 1.7 – Программные конструкции языка BNA-2024

## **Область видимости**

В языке BNA-2024 все переменные являются локальными. Они обязаны находится внутри программного блока функций (по принципу С++). Объявление глобальных переменных не предусмотрено. Объявление пользовательских областей видимости не предусмотрено.

## **Семантические проверки**

Таблица с перечнем семантических проверок, предусмотренных языком, приведена в таблице 1.8.

|  |  |
| --- | --- |
| Номер | Правило |
| 1 | В функции отсутствует return |
| 2 | Тип возвращаемого значения должен совпадать с типом функции при её объявлении |
| 3 | Тип данных передаваемых значений в функцию должен совпадать с типом параметров при её объявлении |
| 4 | В функцию должны быть переданы параметры |
| 5 | Тип данных результата выражения должен совпадать с типом данных идентификатора, которому оно присваивается |
| 6 | Функция не всегда возвращает значение |
| 7 | Попытка объявить функцию в main |
| 8 | Ошибка в блоке main: функция возвращает не целочисленное значение |
| 9 | Количество аргументов передаваемых в функцию |
| 10 | Попытка переопределить библиотечную функцию |

Таблица 1.8 – Семантические проверки

## **Распределение оперативной памяти на этапе выполнения**

Все переменные размещаются в стеке.

## **Стандартная библиотека и её состав**

Стандартная библиотека BNA-2024 написана на языке программирования C++. Функции стандартной библиотеки с описанием представлены в таблице 1.9.

|  |  |  |
| --- | --- | --- |
| Функция(C++) | Возвращаемое значение | Описание |
| unsigned int strLength (char\* str) | uint | Функция определяет длину строки |
| unsigned int toPow(unsigned int a, unsigned int b) | uint | Функция возведения числа в степень |
| unsigned int random(unsigned int a) | uint | Функция вывода на консоль строкового идентификатора/литерала |
| void \_stdcall \_outInt(unsigned long value) |  | Функция вывода на консоль целочисленного идентификатора/литерала |
|  |  |  |

Таблица 1.9 – Состав стандартной библиотеки

## **Ввод и вывод данных**

В языке BNA-2024 не реализованы средства ввода данных.

Для вывода данных в стандартный поток вывода предусмотрен оператор out, который входят в состав стандартной библиотеки и описан в таблице 1.9.

## **Точка входа**

В языке BNA-2024 каждая программа должна содержать главную функцию main, т. е. точку входа, с которой начнется последовательное выполнение программы.

## **Препроцессор**

Препроцессор в языке программирования BNA-2024 не предусмотрен.

## **Соглашения о вызовах**

В языке вызов функций происходит по соглашению о вызовах stdcall. Особенности stdcall:

– все параметры функции передаются через стек;

– память высвобождает вызываемый код;

– занесение в стек параметров идёт справа налево.

## **Объектный код**

BNA-2024 транслируется в язык ассемблера.

## **Классификация сообщений транслятора**

В случае возникновения ошибки в коде программы на языке BNA-2024 и выявления её транслятором в текущий файл протокола выводится сообщение. Их классификация сообщений приведена в таблице 1.10.

|  |  |
| --- | --- |
| Интервал | Описание ошибок |
| 0-99 | Системные ошибки |
| 100-109 | Ошибки параметров |
| 110-119 | Ошибки открытия и чтения файлов |
| 120-129 | Ошибки лексического анализа |
| 600-699 | Ошибки синтаксического анализа |
| 700-800 | Ошибки семантического анализа |

Таблица 1.10. Классификация сообщений транслятора