Министерство науки и высшего образования РФ

Федеральное государственное бюджетное образовательное учреждение высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ

Кафедра экономической математики, информатики и статистики (ЭМИС)

ОСНОВЫ АЛГОРИТМИЗАЦИИ. СРЕДА VISUAL STUDIO. ВВЕДЕНИЕ В ЯЗЫК СИ

Отчёт по лабораторной работе по дисциплине “Введение в программирование”

Студент гр. 543-1

/А.Е. Мухамеджан

“\_\_\_\_”\_\_\_\_\_\_\_\_\_\_\_2023

Руководитель доцент

кафедры ЭМИС

\_\_\_\_\_\_\_\_ Е.А. Шельмина

“\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_2023г.

Томск2023

**Лабораторная работа №2**

**Разработка линейных алгоритмов на языке Си**

Введение: цель работы изучить структуру программы на языке Cи, операторы присваивания, ввода и вывода данных, используемые при составлении программ линейной структуры.

Ход работы: Структура программы на языке Си. Выполнение всех программ, написанных на языке Cи, начинается с функции, именуемой main. Структура программы выглядит следующим образом:

директивы\_препроцессора

int main()

{

определения\_объектов;

исполняемые операторы;

return 0;

}

У функции есть имя (main), после которого в круглых скобках перечисляются аргументы или параметры функции (в данном случае у функции main аргументов нет). У функции может быть результат или возвращаемое значение. Если функция не возвращает никакого значения, то это обозначается ключевым словом void. В фигурных скобках записывается тело функции – действия, которые она выполняет. Оператор return 0; означает, что функция возвращает результат – целое число 0. Вслед за заголовком функции main в фигурных скобках размещается тело функции, которое представляет последовательность определений, описаний и исполняемых операторов. Как правило, определения и описания размещаются до исполняемых операторов. Каждое определение, описание и оператор завершается «;».

Алфавит языка: Множество символов используемых в языке Cи можно разделить на пять групп:

1. Символы, используемые для образования ключевых слов и идентификаторов. В эту группу входят прописные и строчные буквы английского алфавита, а также символ подчеркивания и арабские цифры. Следует отметить, что одинаковые прописные и строчные буквы считаются различными символами, так как имеют различные коды.

Таблица 1 – Символы, используемые для образования ключевых слов и идентификаторов

|  |  |
| --- | --- |
| Прописные буквы латинского алфавита | A B C D E F G H I J K L M N O P Q R S T U V W X Y Z |
| Строчные буквы латинского алфавита | a b c d e f g h i j k l m n o p q r s t u v w x y z |
| Символ подчеркивания | \_ |
| Арабские цифры | 0 1 2 3 4 5 6 7 8 9 |

2. Знаки нумерации и специальные символы. Эти символы используются с одной стороны для организации процесса вычислений, а с другой – для передачи компилятору определенного набора инструкций.

Таблица 2 – Знаки нумерации и специальные символы

|  |  |  |  |
| --- | --- | --- | --- |
| Символ | Наименование | Символ | Наименование |
| , | Запятая | ) | Круглая скобка правая |
| . | Точка | ( | Круглая скобка левая |
| ; | Точка с запятой | } | Фигурная скобка правая |
| : | Двоеточие | { | Фигурная скобка левая |
| ? | Вопросительный знак | < | Меньше |
| ` | Апостроф | > | Больше |
| ! | Восклицательный знак | [ | Квадратная скобка |
| | | Вертикальная черта | ] | Квадратная скобка |
| / | Дробная черта | # | Номер |
| \ | Обратная черта | % | Процент |
| ~ | Тильда | & | Амперсант |
| \* | Звездочка | ^ | Логическое не |
| + | Плюс | = | Равно |
| - | Минус | “ | Кавычки |

3. Управляющие и разделительные символы. К этой группе символов относятся: пробел, символы табуляции, перевода строки, возврата каретки, новая страница и новая строка. Эти символы отделяют друг от друга объекты, определяемые пользователем, к которым относятся константы и идентификаторы. Последовательность разделительных символов рассматривается компилятором как один символ (последовательность пробелов).

4. Кроме выделенных групп символов в языке Cи широко используются так называемые, управляющие последовательности, т.е. специальные символьные комбинации, используемые в функциях ввода и вывода информации. Управляющая последовательность строится на основе использования обратной дробной черты (\) (обязательный первый символ) и комбинацией латинских букв и цифр.

Таблица 3 – Управляющие последовательности

|  |  |  |  |
| --- | --- | --- | --- |
| Управляющая последовательность | Наименование | Управляющая последовательность | Наименование |
| \a | Звонок | \f | Перевод формата |
| \b | Возврат на шаг | \” | Кавычки |
| \t | Горизонтальная табуляция | \’ | Апостроф |
| \n | Переход на новую строку | \0 | Ноль-символ |
| \v | Вертикальная табуляция | \\ | Обратная дробная черта |
| \r | Возврат каретки |

Переменные и константы. Идентификатор: Для символического обозначения величин, переменных, констант, функций и т.п. используются имена или идентификаторы. Идентификаторы в языке Cи – это последовательность знаков, начинающаяся с буквы или знака подчеркивания. В идентификаторах можно использовать заглавные и строчные латинские буквы, цифры и знак подчеркивания. Примеры правильных идентификаторов: abc A12 NameOfPerson BITES\_PER\_WORD

abc и Abc – два разных идентификатора, т.е. заглавные и строчные буквы различаются.

Примеры неправильных идентификаторов:

23X a-b

Ряд слов в языке Cи имеет особое значение и не может использоваться в качестве идентификаторов. Такие зарезервированные слова называются ключевыми.

Список ключевых слов:

|  |  |  |
| --- | --- | --- |
| asm | auto | bad\_cast |
| bad\_typeid | bool | typedef |
| break | case | catch |
| char | class | const |
| const\_cast | continue | default |
| delete | do | double |
| dynamic\_cast | else | enum |
| extern | float | for |
| friend | goto | if |
| inline | int | long |
| mutable | namespace | new |
| operator | private | protected |
| public | register | reinterpret\_cast |
| return | short | signed |
| sizeof | static | static\_cast |
| struct | switch | template |
| then | this | throw |
| try | type\_info | unsigned |
| typeid | union | void |
| using | virtual | xalloc |
| volatile | while |  |

Переменна: программа оперирует информацией, представленной в виде различных объектов и величин. Переменная – это символическое обозначение величины в программе. Как ясно из названия, значение переменной (или величина, которую она обозначает) во время выполнения программы может изменяться.

С точки зрения архитектуры компьютера, переменная – это символическое обозначение ячейки оперативной памяти программы, в которой хранятся данные. Содержимое этой ячейки – это текущее значение переменной.

В языке Cи прежде чем использовать переменную, ее необходимо объявить. Объявить переменную с именем x можно так: int x;

В объявлении первым стоит название типа переменной int (целое число), а затем идентификатор x – имя переменной. У переменной x есть тип – в данном случае целое число.

Основные типы данных: в языке Cи существует несколько стандартных основных типов данных: char short int long float double

Первые четыре типа используются для представления целых, последние два – для представления чисел с плавающей точкой. Переменная типа char имеет размер, естественный для хранения символа на данной машине (обычно, байт), а переменная типа int имеет размер, соответствующий целой арифметике на данной машине (обычно, слово).

Для определения данных целого типа используются различные ключевые слова, которые определяют диапазон значений и размер области памяти, выделяемой под переменные.

|  |  |  |
| --- | --- | --- |
| Тип | Размер памяти в байтах | Диапазон значений |
| char | 1 | От -128 до 127 |
| int | 2 | От -32768 до 32767 |
| short | 2 | От -32768 до 32767 |
| long | 4 | От -2 147 483 648 до 2 147 483 647 |
| unsigned char | 1 | От 0 до 255 |
| unsigned int | 2 | От 0 до 65535 |
| unsigned short | 2 | От 0 до 65535 |
| unsigned long | 4 | От 0 до 4 294 967 295 |

Для переменных, представляющих число с плавающей точкой используются следующие типы: float, double, long double.

Величина типа float занимает 4 байта. Из них 1 байт отводится для знака, 8 бит для избыточной экспоненты и 23 бита для мантиссы. Отметим, что старший бит мантиссы всегда равен 1, поэтому он не заполняется, в связи с этим диапазон значений переменной с плавающей точкой приблизительно равен от 3.14E-38 до 3.14E+38.

Величина типа double занимает 8 бит в памяти. Ее формат аналогичен формату float. Биты памяти распределяются следующим образом: 1 бит для знака, 11 бит для экспоненты и 52 бита для мантиссы. С учетом опущенного старшего бита мантиссы диапазон значений равен от 1.7E-308 до 1.7E+308.

Операторы языка программирования: оператором называется элементарная структурная единица программы. Оператор предназначен как для записи алгоритмических действий по преобразованию данных, так и для задания порядка выполнения других действий. Операторы выполняются в порядке их следования в программе. Операторы отделятся друг от друга точкой с запятой. Операторы делятся на:

− простые (не содержат в себе других операторов);

− составные (включают в себя один или несколько дополнительных операторов).

Присваивание: переменной можно присвоить какое-либо значение с помощью операции присваивания. Присвоить – это значит установить текущее значение переменной. По-другому можно объяснить, что операция присваивания запоминает новое значение в ячейке памяти, которая обозначена переменной

|  |  |
| --- | --- |
| int x; | // объявить целую переменную x |
| int y; | // объявить целую переменную y |
| x = 0; | // присвоить x значение 0 |
| y = x + 1; | // присвоить y значение x + 1, т.е. 1 |
| x = 1; | // присвоить x значение 1 |
| y = x + 1; | // присвоить y значение x + 1, теперь уже 2 |