1. Теория.

1.1. Форматы ввода-вывода.

Оператор format. Помогает запоминать форматы дескриптора. Аналогичные записи:

write(1,’1x,f8.3,i10’) a, i

и

1 format(1x,f8.3,i10)

write(1,1) a, i

Возможно использование в дескрипторе константы или целочисленные выражения используя угловые скобки: <i> или даже <i-j>.

При записи массива одной строкой можно использовать просто write(1,\*) a. Записаны будут все элементы массива а, по порядку. Можно также записать и массивы большей размерности, но тогда написание как придется. Если количество символов в строке задавать через дескриптор, то можно записать правильную структуру. При этом надо учесть что для двумерных массивом по умолчанию первая размерность – количество столбцов а вторая количество строк. Для a(i,j) i – количество столбцов, j – строк.

Возможно делать вывод циклически: write(1,80a1) (‘\_’,k=1,80) выведет 80 нижних подчеркиваний, что даст горизонтальную линию.

Если присутствует несколько элементов 1 типа, которые надо выводить можно вместо дескриптора ‘(i10,i10,i10)’ использовать ‘(3i10)’. Возможны более сложные записи: ‘(3(a1,i10,f8.3),5i10)’ даст три раза подряд последовательность символ, 10 знаков на целое, 8 знаков на дробное, из них 3 знака после запятой, и еще 5 подряд целых на 10 знаков.

Наиболее частые дескрипторы:

Целые: i*w*[.m] – *w* длина поля, m – число ведущих нулей. b и o вместо i для двоичного/восьмеричного представления.

Вещественные: f*w*.d *‑ w* длина поля, d – число цифр после точки. d – для двойной точности, e – для Е-формата.

Логический: l*w*.

Символьный: a[w] – в данном случае длина поля – необязательный параметр. Если он не указан, записывается вся символьная переменная.

Перед выводом данных надо внимательно смотреть тип.

Аналогично дескрипторы используются для ввода данных, с помощью read. Правда, если тип записи в файл неизвестен, лучше использовать \*.

Для дескриптора е возможно указать точное число знаков для определения степени: e12.8e4 – будет 4 знака, к примеру: 456.12345678E+0012.

Так же часто используется формат es*w*.d – называемый научным форматом. В нем число перед точкой всегда лежит в диапазоне от 1 до 10, так число из предыдущего примера будет 4.56123456 E+14.

Помимо этого есть большое число дескрипторов управления, как правило, нужных для красоты вывода.

1.2. Операции с файлами.

Существуют следующие операторы работы с файлами:

|  |  |
| --- | --- |
| backspace | Возвращает файл на одну запись назад. Если записей нет, или достигнут конец файла прошлой записью, то ничего не меняется.  backspace f |
| rewind | Перемещает на начало файла.  rewind f |
| endfile | Делает запись конец файла  endfile f |
| open | Создает устройство ввода/вывода и присоединяет к нему файл. Имеет большое количество необязательных опций которые можно изучить самостоятельно. |
| close | Отсоединяет файл от устройства в/в и закрывает это устройство. После этого можно использовать это устройство повторно с новым или тем же файлом.  close(f)  бывает удобно сразу удалить файл, если он временный:  close(f,status=’delete’) |
| read | Осуществляет передачу из файла. |
| write | Осуществляет передачу в файл. |
| print | Выводит данные не экран.  print \*, f или print ‘(i10)’, f |
| inquire | Возвращает свойства устройства или файла. Вариантов свойств выводимых много и все почти специфичны. Наиболее интересны:  inquire(1, name) name – возвращает имя файла, присоединенного к устройству вместе с путем.  inquire(iolength=il) a, s – il – целая, возвращает размер списка вывода. Это даст представление о том какой размер нужен для вывода переменных или массивов a и s. |
| eof | Возвращает правду если в файле запись конец файла.  flac=eof(f) |

2. Задание.

1.1. Форматы ввода-вывода.

Показать одновременный или последовательный вывод в файл с разными дескрипторами. Показать ввод данных с дескрипторами и без.

1.2. Операции с файлами.

Показать основные операции с файлами.