МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ДЕРЖАВНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД

**«НАЦІОНАЛЬНИЙ ГІРНИЧИЙ УНІВЕРСИТЕТ»**

**Кафедра програмного забезпечення комп'ютерних систем**

**Лабораторна робота №1**

з дисципліни: «ЯПЗ»

|  |  |
| --- | --- |
| Виконав: | ст. групи ПЗм-15-1м |
| Перевірив: | Козлов В.П. |

Дніпропетровськ

2016

**Тема:** Метрики Холстеда

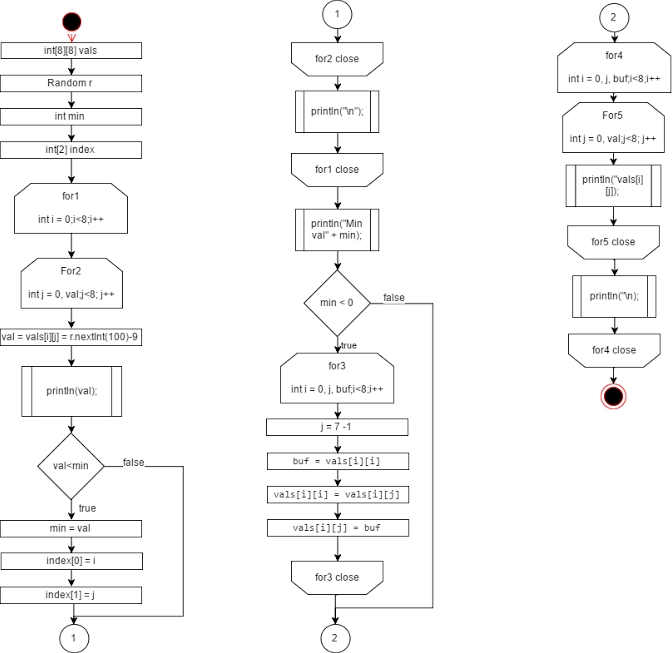
**Цель:** расчет метрик Холстеда для разработанной программы с результатами, представленными в виде таблицы.

**Задание:** 9. Ввести массив А(8, 8). Если минимальный элемент данного массива отрицателен, поменять местами главную и побочную диагонали массива. Вывести минимальный элемент, номера строки и столбца, в которых он находится, исходный и результирующий массивы.

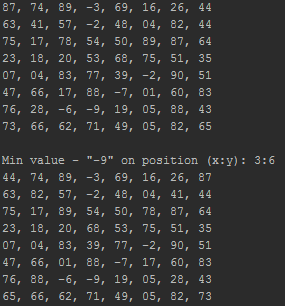
**Исходный текст программы:**

//Массив исходных значений (матрица 8х8)  
**int**[][] vals = **new int**[8][8];  
Random r = **new** Random();  
**int** min = **Integer**.MAX\_VALUE;  
//Массив индексов минимального значения массива  
**int**[] index = **new int**[2];  
//Наполнение исходного/результирующего массивов  
//Поиск в массиве минимального значения  
**for** (**int** i = 0; i **<** 8; i**++**) {  
 **for** (**int** j = 0, val; j < 8; j++) {  
 val = vals[i][j] = r.nextInt( 100 ) - 9;  
 **System.out.print**(  
 ((val < 10 && val >= 0) ? ("0" + val) : val)  
 + (j == 7 ? "" : ", ") );  
 **if** ( val < min ) {  
 min = val;  
 //сохранение указателей на минимальное значение  
 index[0] = i;  
 index[1] = j;  
 }  
 }  
 **System.out.println**();  
}  
//Вывод минимального значения с указателями на него  
**System.out.println**( "\nMin value - \"" + min + "\" on position (x:y): " + index[1] + ":" + index[0] );  
  
**if** ( min < 0 ) {  
 **for** (**int** i = 0, j, buf; i < 8; i++) {  
 j = 7 - i; // указатель для побочной диагонали  
 buf = vals[i][i];  
 vals[i][i] = vals[i][j];  
 vals[i][j] = buf;  
 }  
}  
//Вывод результирующего массива  
**for** (**int** i = 0; i < 8; i++) {  
 **for** (**int** j = 0; j < 8; j++) {  
  **System.out.print**(  
 (vals[i][j] < 10 && (vals[i][j] >= 0) ? ("0" + vals[i][j]) : vals[i][j])  
 + (j == 7 ? "" : ", ") );  
 }  
 **System.out.println**();  
}

**Схема алгоритма:**



**Копии экрана с результатами работы программы:**



**Описание программы**:

Задача программы выполнить поиск минимального значения в матрице 8\*8. В случае если минимальное значение меньше 0, то необходимо переместить значения основной диагонали матрицы в побочную, и наоборот. Для этого выполнение программы можно разделить на 3 этапа:

1. Объявление и инициализация исходных данных, а именно переменные исходного массива, вспомогательные переменные для хранения минимального значения и его индексов.
2. Ввод значений матрицы и проверка введенного значения для поиска минимального. Вывод введенных данных на экран и найденного минимального значения.
3. Перемещение диагоналей матрицы в случае если найденное минимальное значение меньше нуля. Вывод результирующей матрицы.

**Расчет метрик Холстеда:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ***f***j | Оператор | ***f***1j |  |  | ***fi*** | Операнд | ***f***2i |
| 1 | = | 20 |  |  | 1 | vals | 10 |
| 2 | + | 9 |  |  | 2 | r | 2 |
| 3 | ++ | 5 |  |  | 3 | min | 5 |
| 4 | < | 9 |  |  | 4 | i | 22 |
| 5 | >= | 2 |  |  | 5 | index | 5 |
| 6 | for | 5 |  |  | 6 | j | 18 |
| 7 | if | 2 |  |  | 7 | buf | 3 |
| 8 | int | 10 |  |  | 8 | val | 8 |
| 9 | new | 3 |  |  |  | | 73 |
| 10 | else | 1 |  |  |  |  |  |
| 11 | return | 1 |  |  |  |  |  |
| 12 | ; | 29 |  |  |  |  |  |
| 13 | ( ) | 19 |  |  |  |  |  |
| 14 | , | 3 |  |  |  |  |  |
| 15 | == | 2 |  |  |  |  |  |
| 16 | { } | 7 |  |  |  |  |  |
| 17 | print | 5 |  |  |  |  |  |
| 18 | [ ] | 28 |  |  |  |  |  |
| 19 | : | 5 |  |  |  |  |  |
|  | | 165 |  |  |  |  |  |

**Словарь программы: n = 19 + 8 = 27**

**Длинна программы: N = 165 + 73 = 238**

**Объем программы: V = 238 log2 27 = 1131.66**

**Вывод:** в ходе выполнения лабораторной работы, согласно условиям задачи, была написана программа, исходный код которой был проанализирован согласно метрик Холстеда, в результате был выведен коэффициент объема программы - 1131.66.