Министерство образования Республики Беларусь

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

**Язык программирования “PERL”**

**Метрики Маккейба, Джилба и граничных значений.**

|  |  |
| --- | --- |
| Выполнил: | студенты группы 772302  Углянец А.В., Нагула А.А. |
| Проверил: | Болтак С.В. |

Минск 2018

**Метрика Джилба**

Суть метрики Джилба заключается в определении логической сложности программы как насыщенности программы условными операторами **IF–THEN–ELSE**. Обычно используются два вида метрики Джилба:

* CL — количество условных операторов, характеризующее абсолютную сложность программы;
* cl — насыщенность программы условными операторами, характеризующая относительную сложность программы;

cl = ,

где ƞ – общее количество операторов.

Расширением метрики Джилба является максимальный уровень вложенности условного оператора **CLI**.

**Код программы:**

sub game

{

$nextFigure=$figures[int rand @figures];

Tetris::UI::showGlass;

while()

{

$figure=$nextFigure;

$nextFigure=$figures[int rand @figures];

Tetris::UI::showNextFigure($nextFigure);

Tetris::UI::showScore($score);

$figureX=int((GLASS\_COLS-$figure->width)/2);

$figureY=0;

term unless canStay;

while()

{

if(Tetris::UI::inputPending(5))

{

my $key=Tetris::UI::getKey();

if($key==0x101) # UP

{

showMove(ROTATE\_LEFT);

}

elsif($key==0x102) # DOWN

{

showMove(ROTATE\_RIGHT);

}

elsif($key==0x103) # LEFT

{

showMove(MOVE\_LEFT);

}

elsif($key==0x104) # RIGHT

{

showMove(MOVE\_RIGHT);

}

elsif($key==0x20) # SPACE

{

fall;

}

}

elsif(canMove(MOVE\_DOWN)) { showMove(MOVE\_DOWN); }

else

{

addFigure;

last;

}

}

}

}

Для нашей программы имеем следующие значения:

* CL = 9;
* cl = 9 / 19 = 0,47;
* CLI = 8;

**Метрика граничных значений**

Для оценки сложности программы с использованием метрики граничных значений граф G разбивается на максимальное число подграфов, удовлетворяющих следующим условиям:

* вход в подграф осуществляется через вершину выбора;
* каждый подграф включает вершину (нижнюю границу подграфа), в которую можно попасть из любой другой вершины подграфа;
* число вершин, образующих подграф, равно скорректированной сложности вершины выбора;
* каждая принимающая вершина имеет скорректированную сложность, равную 1;
* конечная вершина имеет скорректированную сложность, равную 0.

**Таблица 1.** Свойства подграфов программы

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Свойства подграфов программы | Номер вершины выбора | | | | | | |
| 3 | 4 | 8 | 10 | 12 | 14 | 16 |
| Номера вершин перехода | 4,5 | 6,7 | 9,10 | 11,12 | 13,14 | 15,16 | 17 |
| Скорректированная сложность вершины выбора | 14 | 3 | 10 | 8 | 6 | 4 | 2 |
| Номера вершин подграфа | 4,5,6,7,  9,10,11,12,13,  14,15,16,17 | 6,7 | 9,10,11,12,13,  14,15,16,17 | 11,12,13,14,  15,16,17 | 13,14,15,  16,17 | 15,16,17 | 17 |
| Номер нижней границы подграфа | 18 | 18 | 18 | 18 | 18 | 18 | 18 |

**Таблица 2.** Скорректированные сложности вершин графа программы

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Номер вершины графа  программы |  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
| Скорректированная  сложность  вершины графа |  | 1 | 1 | 14 | 3 | 1 | 1 | 1 | 10 | 1 | 8 | 1 | 6 | 1 | 4 | 1 | 2 | 1 | 1 | 0 |

*Sa* = 58

*So*= 1 – (19 – 1)/58 = 0,69.

Абсолютная сложность 28

Относительная сложность 28/760=0.037