

Алгоритм, программа и пример расчета временных параметров сетевого графика на Visual Basic, встроенном в Microsoft Excel

1. Обозначения в программе

В Visual Basic можно использовать только латинские буквы и нет обозначений с индексами. Поэтому обозначения в программе отличаются от обычно применяемых в учебниках:

- n – число работ в сетевом графике
- NN – число заданных работ в заполняемой таблице
- $N1(i)$ – номер события начала i -й работы
- $N2(i)$ – номер события окончания i -й работы
- t_{\max} – максимальная продолжительность выполнения всех работ (длина критического пути)
- $t(i)$ – продолжительность i -й работы
- $t1(i)$ – раннее начало i -й работы
- $t2(i)$ – раннее окончание i -й работы
- $t3(i)$ – позднее начало i -й работы
- $t4(i)$ – позднее окончание i -й работы
- $R1(i)$ – общий резерв времени i -й работы
- $R2(i)$ – частный резерв времени i -й работы

2. Ввод исходных данных

Расчет сетевого графика выполняется в табличной форме, где строки соответствуют работам, а в столбцах записываются вычисленные для этих работ величины ранних и поздних начал, окончаний работ и резервы времени.

На листе Excel заполняется таблица номеров событий начала работ, номеров событий окончания работ продолжительность каждой работы. События нумеруются арабскими цифрами, начиная с 1 без пропусков.

Работы записываются в таблицу по мере возрастания номеров начал работ. Сначала записывают все работы, началом которых является исходное событие 1, затем все работы, имеющие началом событие 2, и так далее. В расчетной программе требуется, чтобы таблица заканчивалась пустой (незаполненной) строкой.

Работы располагаются в порядке возрастания номеров начал работ, а при одинаковых номерах начала – в порядке возрастания номеров окончания этих работ.

При этом можно воспользоваться встроенной в Excel функцией сортировки по столбцам номеров начала и окончания работ.

В программном коде Visual Basic записан специальный блок, переставляющий введенные работы в порядке возрастания номеров.

3. Последовательность вычислений, реализуемая в программном коде Visual Basic

3.1. Определяется число заданных работ

Это нужно, чтобы задать размер массивов данных. Следует обратить внимание, что для этого используется цикл с условием окончания по наличию пустой (незаполненной) ячейки. При этом учитывается, что проверка окончания цикла выполняется после увеличения числа n на единицу.

```
n = 0  
Do  
n = n + 1: NN = Cells(n + 1, 2)  
Loop While NN <> 0  
n = n - 1
```

3.2. Объявление массивов данных с числом элементов, равных числу найденных работ

```
ReDim N1(n), N2(n), t(n), t1(n), t2(n), t3(n), t4(n), R1(n), R2(n)
```

3.3. Ввод данных с листа Excel с помощью цикла

```
For i = 1 To n  
N1(i) = Cells(i + 1, 2): N2(i) = Cells(i + 1, 3): t(i) = Cells(i + 1, 4)  
Next i
```

3.4. Перестановка работ в порядке возрастания номеров с помощью циклов

```
For i = 1 To n - 1  
For j = i + 1 To n  
If N1(j) < N1(i) Then  
x = N1(i): y = N2(i): Z = t(i)  
N1(i) = N1(j): N2(i) = N2(j): t(i) = t(j)  
N1(j) = x: N2(j) = y: t(j) = Z  
End If  
Next j  
Next i
```

```
For i = 1 To n - 1
```

```

For j = i + 1 To n
If N1(j) = N1(i) Then
x = N1(i): y = N2(i): Z = t(i)
If N2(j) < N2(i) Then
N1(i) = N1(j): N2(i) = N2(j): t(i) = t(j)
N1(j) = x: N2(j) = y: t(j) = Z
End If
End If
Next j
Next i

```

3.5. Вывод перенумерованных работ на лист Excel с помощью цикла

```

For i = 1 To n
Cells(i + 1, 2) = N1(i): Cells(i + 1, 3) = N2(i): Cells(i + 1, 4) = t(i)
Next i

```

3.6. Расчет ранних начал и ранних окончаний всех работ

Ранние начала вычисляются от исходного события 1 к завершающему событию n так, чтобы для расчета ранних начал последующих работ были рассчитаны ранние окончания всех предшествующих работ.

Для всех работ, имеющих началом исходное событие 1, раннее начало $t1(i) = 0$

Для остальных работ, имеющих номера событий $N1(i) \sim N2(i)$, раннее начало равно максимальному значению из ранних окончаний всех предшествующих работ $N1(j) \sim N2(j)$, имеющих ранним окончанием то же событие.

Для этого производится поиск всех предшествующих работ, имеющих окончанием то же событие, что начало данной работы.

То есть, если $t2(j) > t1(i)$ при $N1(i) = N2(j)$, то $t1(i) = t2(j)$.

Ранние окончания всех работ вычисляются по формуле

$$t2(i) = t1(i) + t(i)$$

В программном коде для этого используется цикл

```

For i = 1 To n
t1(i) = 0: t2(i) = t1(i) + t(i)
Next i
For i = 1 To n
For j = 1 To n
If N1(i) = N2(j) And t2(j) > t1(i) Then t1(i) = t2(j): t2(i) = t1(i) + t(i)
Next j
Next i

```

3.7. Определение максимальной продолжительности выполнения всех работ (длины критического пути сетевого графика)

Для всех работ, имеющих окончанием завершающее событие n , позднее окончание равно максимальному значению из ранних окончаний всех предшествующих работ и вычисляется по формуле

$$t4(i) = \max t2(i)$$

Максимальная продолжительность выполнения всех работ (длина критического пути сетевого графика) равна максимальному значению из всех поздних окончаний работ, имеющих окончанием завершающее событие n

$$t_{\max} = \max t4(i) = \max t2(i)$$

В программном коде для этого используется цикл

```
tmax = 0  
For i = 1 To n  
    If t2(i) > tmax Then tmax = t2(i)  
Next i
```

3.8. Расчет поздних начал и поздних окончаний всех работ

Поздние окончания вычисляются от завершающего события n к начальному событию 1 так, чтобы для расчета поздних окончаний предшествующих работ были рассчитаны поздние начала всех последующих работ.

Для всех работ, имеющих окончанием завершающее событие n , позднее окончание

$$t4(i) = t_{\max}$$

Для остальных работ, имеющих номера событий $N1(i) \sim N2(i)$, позднее окончание равно минимальному значению из поздних начал всех последующих работ $N1(j) \sim N2(j)$, имеющих началом то же событие.

Для этого производится поиск всех последующих работ, имеющих началом то же событие, что окончание данной работы.

То есть, если $t3(j) < t4(i)$ при $N2(i) = N1(j)$, то $t4(i) = t3(j)$.

Поздние начала всех работ вычисляются по формуле

$$t3(i) = t4(i) - t(i)$$

В программном коде для этого используется цикл

```
For i = 1 To n  
    t4(i) = tmax: t3(i) = t4(i) - t(i)  
Next i
```

```

For i = n To 1 Step -1
For j = n To 1 Step -1
If N2(i) = N1(j) And t3(j) < t4(i) Then t4(i) = t3(j): t3(i) = t4(i) - t(i)
Next j
Next i

```

3.9. Расчет резервов времени всех работ

Общий резерв времени в сетевом графике – это максимальное время, на которое можно перенести начало работы или увеличить ее продолжительность без увеличения продолжительности критического пути.

Частный резерв времени в сетевом графике – это время, на которое можно перенести начало работы или увеличить ее продолжительность без изменения раннего начала последующих работ.

Расчет выполняется от завершающего n к начальному событию 1.

Общий резерв времени работы $N1(i) \sim N2(i)$ равен разности ее поздних и ранних начал или окончаний.

Частный резерв времени этой же работы равен разности между ранним началом последующей работы $N1(j) \sim N2(j)$ и ранним окончанием данной работы $N1(i) \sim N2(i)$ (кроме работ, имеющих окончанием завершающее событие).

Для всех работ, имеющих окончанием завершающее событие, частный резерв времени равен общему резерву.

В программном коде для этого используется цикл

```

For i = n To 1 Step -1
R1(i) = t4(i) - t2(i): R2(i) = R1(i)
For j = n To 1 Step -1
If N2(i) = N1(j) Then R2(i) = t1(j) - t2(i)
Next j
Next i

```

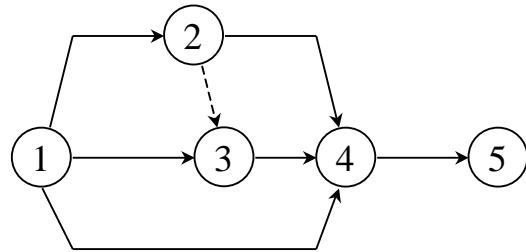
3.10. Вывод результатов на лист Excel с помощью цикла

```

For i = 1 To n
Cells(i + 1, 1) = i:
Cells(i + 1, 5) = t1(i): Cells(i + 1, 6) = t2(i)
Cells(i + 1, 7) = t3(i): Cells(i + 1, 8) = t4(i)
Cells(i + 1, 9) = R1(i): Cells(i + 1, 10) = R2(i)
Next i

```

Пример расчета сетевого графика



1. Заполнить столбцы, выделенные цветом. Таблица должна заканчиваться незаполненной (пустой) строкой

номер работы	номер начала	номер конца	продолжительность	раннее начало	раннее окончание	позднее начало	позднее окончание	общий резерв	частный резерв
1	2	9							
1	3	6							
1	4	4							
2	3	0							
2	4	4							
3	4	4							
4	5	2							

2. Щелкнуть кнопку "Выполнить расчет временных параметров сетевого графика"

Выполнить расчет временных параметров сетевого графика

номер работы	номер начала	номер конца	продолжительность	раннее начало	раннее окончание	позднее начало	позднее окончание	общий резерв	частный резерв
1	1	2	9	0	9	0	9	0	0
2	1	3	6	0	6	3	9	3	3
3	1	4	4	0	4	9	13	9	9
4	2	3	0	9	9	9	9	0	0
5	2	4	4	9	13	9	13	0	0
6	3	4	4	9	13	9	13	0	0
7	4	5	2	13	15	13	15	0	0

Программный код (текст программы) Visual Basic, встроенного в Microsoft Excel для расчета сетевого графика

```
Private Sub CommandButton1_Click()
```

```
'Определение числа заданных работ
n = 0
Do
n = n + 1: NN = Cells(n + 1, 2)
Loop While NN <> 0
n = n - 1
```

```
'Ввод исходных данных с листа Excel  
ReDim N1(n), N2(n), t(n), t1(n), t2(n), t3(n), t4(n), R1(n), R2(n)  
For i = 1 To n  
    N1(i) = Cells(i + 1, 2): N2(i) = Cells(i + 1, 3): t(i) = Cells(i + 1, 4)  
Next i
```

```
'Сортировка работ по возрастанию номеров начала и конца  
For i = 1 To n - 1  
    For j = i + 1 To n  
        If N1(j) < N1(i) Then  
            x = N1(i): y = N2(i): Z = t(i)  
            N1(i) = N1(j): N2(i) = N2(j): t(i) = t(j)  
            N1(j) = x: N2(j) = y: t(j) = Z  
        End If  
    Next j  
Next i
```

```
For i = 1 To n - 1  
    For j = i + 1 To n  
        If N1(j) = N1(i) Then  
            x = N1(i): y = N2(i): Z = t(i)  
        If N2(j) < N2(i) Then  
            N1(i) = N1(j): N2(i) = N2(j): t(i) = t(j)  
            N1(j) = x: N2(j) = y: t(j) = Z  
        End If  
    End If  
Next j  
Next i  
For i = 1 To n  
    Cells(i + 1, 2) = N1(i): Cells(i + 1, 3) = N2(i): Cells(i + 1, 4) = t(i)  
Next i
```

```
'Определение ранних начал и ранних окончаний  
For i = 1 To n  
    t1(i) = 0: t2(i) = t1(i) + t(i)  
Next i  
For i = 1 To n  
    For j = 1 To n  
        If N1(i) = N2(j) And t2(j) > t1(i) Then t1(i) = t2(j): t2(i) = t1(i) + t(i)  
    Next j  
Next i
```

```
'Определение максимальной продолжительности  
tmax = 0  
For i = 1 To n  
    If t2(i) > tmax Then tmax = t2(i)  
Next i
```

```
'Определение поздних начал и поздних окончаний  
For i = 1 To n  
    t4(i) = tmax: t3(i) = t4(i) - t(i)  
Next i  
For i = n To 1 Step -1  
    For j = n To 1 Step -1  
        If N2(i) = N1(j) And t3(j) < t4(i) Then t4(i) = t3(j): t3(i) = t4(i) - t(i)  
    Next j  
Next i
```

```
'Определение резервов времени
```

```
For i = n To 1 Step -1
    R1(i) = t4(i) - t2(i): R2(i) = R1(i)
    For j = n To 1 Step -1
        If N2(i) = N1(j) Then R2(i) = t1(j) - t2(i)
    Next j
Next i

'Вывод результатов на лист Excel
For i = 1 To n
    Cells(i + 1, 1) = i:
    Cells(i + 1, 5) = t1(i): Cells(i + 1, 6) = t2(i)
    Cells(i + 1, 7) = t3(i): Cells(i + 1, 8) = t4(i)
    Cells(i + 1, 9) = R1(i): Cells(i + 1, 10) = R2(i)
Next i

End Sub
```