2015 International Mathematical Modeling Challenge

Submitted Solution

Team Control Number

2015005

The attached document is a solution submitted in response to the 2015 International Mathematical Modeling Challenge (IM^{2}C) problem **'Move Scheduling'** and is provided as a reference point for participants in forthcoming IM^{2}C events.

Please take great care to ensure that no part of this solution is copied or plagiarised in the preparation of your solution for this year's International Mathematical Modeling Challenge.

www.immchallenge.org.au

For office use only	Team Control Number	For office use only
T1 ________________	**2015005**	F1 ________________
T2 ________________		F2 ________________
T3 ________________		F3 ________________
T4 ________________		F4 ________________

2015

The International Mathematical Modeling Challenge (IM^{2}C)
Summary Sheet
(Attach a copy of this page to the front of your solution paper.)

Abstract

Scheduling is an important part of shooting a film. In this paper, we established a model for movie scheduling and implemented a program in Visual Basic which allows the producers to output a detailed schedule by inputting all the *Constraints* including the available time of movie stars, filming sites and special props.

Looking through the list of factors, we have observed that all these factors can be viewed as one type of *Constraints* and can be modeled in the same way.

For Question 1, we combined all different types of initial conditions and represented them by matrices. Then we reasonably weighted the *Footage Matrices* to form the *Schedule Matrix*, eventually obtaining an optimal schedule.

For Question 2, the only thing we need to do is to apply the altered *Constraints* as new initial conditions into to the model provided in Question 1, and then generate a new schedule accordingly.

For Question 3, we defined an importance index for each of the *Constraints*. Through data analysis, statistics and charts we concluded that the most important *Constraint* is the one with the highest importance index, and it will change based on the different type of the film.

Through the random simulation of the model, we also verified the stability and efficiency of the model, knowing that the average filming time would be under 28-unit time.

Finally, we have analyzed the advantages and disadvantages of the model and proposed two improvements which will optimized the mode further

Key words: *Constraints*, Matrix, Weight, Schedule, Random Simulation

Contents

1 Problem Description

1.1 Introduction

Nowadays, the pace of people's life is getting faster and faster. The fiercely competitive of study and work has caused more and more stress. There is no doubt that watching a movie, as a way of entertainment that is suitable for all ages, has become the one of the best ways for people to relax. The two-hour time is enough for us to temporarily forget the pain or boredom in the reality. However, few people might think of how much time it would cost to produce a movie. While the special effects are spectacular and the scenes are beautiful, the filming of these scenes is never as easy as what you see in the cinema. To produce a successful film, the controlling of every session in filming and achieving the situation with the luck, advances and harmony, is the most important thing, since any session in the filming will impact the movie production schedule and final releasing date. This paper intends to establish a mathematical model to help the producers to solve these problems.

1.2 Origin of the Problem

A large movie production studio requests us to make a filming schedule by building a mathematical model, and to provide examples and test results to convince them of the model's effectiveness.

Question 1:

There are 5 main factors which should be taken into consideration:

i. The schedule of the movie stars;
ii. The available time of the filming sites;
iii. The time for scenery and props to be prepared;
iv. The available time of the special props
v. Some scenes need to be processed by the computer or be filmed after the physical system is finished.

Besides, we should leave enough time for some possible re-filming after the cutting.

Question 2:

The model should be able to adjust the schedule when one of the factors changes.

Question 3:

The model mentioned in the Question 1 should be able to indicate the key factor affecting the schedule, which can cause the most serious delay if an accident happens to it.

2 Analysis of the Problem

The key to the question is the arrangement of the filming of various *Footages*. Since the preparation and the editing is relatively fixed for a certain movie, we should focus on how to arrange the filming order properly.

Question 1: We can view all the factors in the same way and make a *Filming Matrix* with all the constraints provided including the availability of movie stars, filming sites and special props.

We can then determine the order of filming on certain *Footages* (at the correct time but not always the first) by appropriately weighting the *Footages*, and generate the optimal schedule using the VB program implemented according to the model.

Question 2: If any *Constraint* changes (may be caused by the injury of stars, the changing of the weather on the outdoor scene, and damage of some special props, etc.), we can substitute the altered *Constraints* into the model in Question 1 we will get a new schedule that will satisfy the new conditions.

Question 3: We defined a universal importance index for every *Constraint*. Among all the factors, including the movie stars, filming sites and special props, the most important one is the one with the highest importance index.

3. Statement of the Symbols

Symbol	Description	Remark
n	the estimated number of time units for filming	an input value big enough for the filming of the movie
r_i	*Constraints*	
A_{r_i}	*Constraint Matrix*	a 0-1 matrix with one row and n columns
m	the number of *Constraints*	
$\boldsymbol{F}_k$	*Footage-Constraint Set*	a set with q_k *Constraint*
q_k	the number of *Constraints* needed for *Footage K*	
$\boldsymbol{B}_k$	*Footage Matrix*	a 0-1 matrix with one row and n columns
p	the number of *Footages*	
$\boldsymbol{C}$	*Filming Matrix*	a 0-1 matrix with p rows and n columns
λ	weight of a *Footage*	
$\boldsymbol{E}$	*Schedule Matrix*	a 0-1 matrix with one row and n columns
y_1,y_2,y_3	movie stars	
z_1,z_2	filming sites	
w_1,w_2	special props	

4 Assumption

4.1 Preparation

i. We assume that the time of preparation is fixed
ii. We assume that the director is familiar with the local weather, and he can avoid the extremely bad weather (such as the rain season) which would influence the outdoor filming too seriously.
iii. We do not take the force majeure (natural disaster like typhoon, flood, hail, or the government act as levying, the society incident, such as strike or riot) into account.

4.2 During the Filming

i. We assume that we can edit the previous *Footage* while continuing filming.
ii. We assume that the indoor scene can be arranged during the filming of the previous *Footage*, so the first scene to film cannot be an indoor one.
iii. We assume that during the filming of each *Footage* the *Constraints* of it stay the same.
iv. Considering that the director should be always present at the filming site, so only one *Footage* could be filmed during the same time unit.
v. We assume that the director won't change the script during filming.
vi. We assume that the crews are experienced, so that they can definitely complete a *Footage* within one time unit.
vii. We don't take the traveling time from one filming site to another (including applying for a visa if in foreign countries) into account.
viii. We assume that only a few stars are restrained by their own schedule, while the others are available at any time.
ix. We don't take the influence of psychological status of the stars into account.
x. We assume the film lasts 100 minutes.
xi. We define that the shooting time needed for each *Footage* is one time unit.
xii. We assume that the whole movie will be cut into 20 *Footages* evenly, so each *Footage* lasts 5 minutes. (In order to calculate more easily, the examples in this paper will only use 7 *Footages* as an illustration).
xiii. In order to simplify the calculation, we assume that one time unit equals to a week.
xiv. We don't take the time of dressing up into account.
xv. We assume that the availability ration of the indoor scene is quite stable, so here we ignore the influence caused by its changes.

4.3 Post-Filming Editing

i. We assume that the time needed for editing is fixed.
ii. We assume that the time needed to re-film after editing is fixed, since the crews are experienced.

5 Model Establishment and Adjustment

5.1 Question 1

5.1.1 Determination of the Method.

By observing the question, the requirement is that when every *Constraint* is known, we need to make a detailed schedule in order to inform the director.

Here are the initial conditions:

i. The number of movie stars (The actors/actresses who cannot join the shooting in a certain period of time due to their own schedule.).

ii. The available time of each movie star.

iii. The number of filming sites (including indoor scenes and outdoor ones)

iv. The available time of each filming site.

v. The number of special props.

vi. The available time of each special prop.

vii. The number of *Footages*

viii. The *Constraints* needed in each *Footage.*

5.1.2 Establishment of the Method

5.1.2.1 Definition

i. *Constraint Matrix*

We see movie stars, filming sites and special props as *Constraints* and call them $r_1, r_2, ..., r_m$

Define $A_{r_i} = (a_{r_i1}, a_{r_i2}, ..., a_{r_in})$ as the *Constraint Matrix*,

in which

$$a_{r_ij} = \begin{cases} 1 & \text{Constraint } r_i \text{ can be shot in week j} \\ 0 & \text{Constraint } r_i \text{ cannot be shot in week j} \end{cases} \quad i = 1,2,3,...,m \qquad j = 1,2,3,...,n$$

ii. *Footage-Constraint Set*

Suppose that the *Constraints* of every *Footage* is $r_{k1}, r_{k2}, ..., r_{kq_k}$

Then define $F_k = \{r_{k1}, r_{k2}, ..., r_{kq_k}\}$ as a *Footage-Constraint Set*

In which $r_{k1}, r_{k2}, ..., r_{kq_k} \in \{r_1, r_2, ..., r_m\}$

iii. *Footage Matrix*

Define $B_k = (b_{k1}, b_{k2}, ..., n_{kn})$ as the *Footage Matrix*

In which

$$b_{kh} = \begin{cases} 1 & \textit{Footage } k \text{ can be shot in the week h} \\ 0 & \textit{Footage } k \text{ cannot be shot in the week h} \end{cases} \quad k = 1,2,3,...,p \qquad h = 1,2,3,...,n$$

And

$$b_{kh} = \prod_{t=1}^{q_k} a_{r_{kt}h}$$

iv. *Filming Matrix*

Define $C=\begin{pmatrix} b_{11} & \dots & b_{1n} \\ \dots & \dots & \dots \\ b_{p1} & \dots & b_{pn} \end{pmatrix}\begin{matrix} \to B_1 \\ \to B_k \\ \to B_p \end{matrix}$ as the *Filming Matrix*. It is a 0-1 matrix formed by arranging the *p Footage Matrices* from top down.

v. *Schedule Matrix*

The requirement is to choose '1's from the *Filming Matrix* in a way that there's one and only one '1' chosen in each row and at most one '1' in each column.

The *Schedule Matrix* should be a matrix with one rows and *n* columns, and the time needed for filming a complete movie is the position of the last non-zero element in the matrix.

We define $E=(e_1,e_2,\dots,e_n)$ as the *Schedule Matrix*.

$$e_j=\begin{cases} k & \text{Footage } k \text{ is to be shot at the week j} \\ 0 & \text{Nothing is to be shot at the week j} \end{cases} \quad j=1,2,\dots,n \quad k\in\{1,2,\dots,p\}$$

5.1.2.2 Rationality Explanations

i. *Constraint Matrix*

We represent the available time of movie stars (or filming sites or special props) as a 0-1 matrix in which every element represents the availability of a movie star (or filming sites or special props) in a particular week. For example, (1,1,1,0,1,1,1,1) shows a particular star is available in the week 1, 2, 3, 5, 6, 7, and 8.

As we can see clearly, all movie stars, filming sites and special props can be represented in the same way, and we call them *Constraints* as a whole.

For example:

Star 1 (*Constraint 1*) $A_{r_1}=(1,1,1,0,1,1,1,1)$

Star 2 (*Constraint 2*) $A_{r_2}=(0,1,1,1,0,1,1,1)$

Star 3 (*Constraint 3*) $A_{r_3}=(1,0,0,1,1,1,1,0)$

Outdoor scene (*Constraint 4*) $A_{r_4}=(1,1,1,0,1,0,1,1)$

Indoor scene (*Constraint 5*) $A_{r_5}=(0,0,1,0,1,1,1,1)$

Special prop (*Constraint 6*) $A_{r_6}=(0,0,1,0,1,0,1,0)$

(Considering that when the movie first start filming, the setting of the indoor scene may have not been completed yet, so in *Constraint 5* the first week is set to be '0'. e.g. A_{r5})

ii. *Footage-Constraint Set*

A certain *Footage* will have its corresponding *Constraints* and during filming of this certain *Footage*, the *Constraints* will not be changed.

For example

Suppose that the shooting of *Footage 2* needs movie star 2 (*Constraint 2*), filming site 2 (*Constraint 5*) and special prop (*Constraint 6*), then the *Footage-Constraint Set* of the *Footage 4* can be described as $F_2 = \{r_2, r_5, r_6\}$

iii. *Footage Matrix*

If one column in $\boldsymbol{B}_k$ is '1', then every *Constraint* in the same column has to be '1'.

For example

$$A_{r_2} = (0,1,1,1,0,1,1,1)$$
$$A_{r_5} = (0,0,1,0,1,1,1,1)$$
$$A_{r_6} = (0,0,1,0,1,0,1,0)$$
$$B_2 = (0,0,1,0,0,0,1,0)$$

We can see clearly that if a column in $\boldsymbol{B}_k$ is '1' then the corresponding value in $\boldsymbol{A}_{r1}$, $\boldsymbol{A}_{r5}$ and $\boldsymbol{A}_{r6}$ must be '1', which means that $b_{kh} = 0$ if and only if $\prod_{t=1}^{q_k} a_{r_{kt}h} = 0$

iv. *Filming Matrix*

For example:

$$C = \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} \begin{matrix} \rightarrow B_1 \\ \rightarrow B_2 \\ \rightarrow B_3 \\ \rightarrow B_4 \\ \rightarrow B_5 \\ \rightarrow B_6 \\ \rightarrow B_7 \end{matrix}$$

It is a 0-1 matrix formed by arranging the 7 *Footage Matrices* from top down.

v. *Schedule Matrix*

Every row represents a *Footage*, so one '1' has to be chosen from each row; every column represents a week and only one *Footage* can be shot in a week at most, so one at most '1' can be chose from one column.

For example

$$E = (7,4,2,0,1,5,3,6)$$

We can conclude that 8 weeks is needed to finish the film, as shown in Table 1 (Every element in $\boldsymbol{E}$ matches the 8 weeks one by one, so we can get the schedule as the following.)

week	1	2	3	4	5	6	7	8
Footage No.	7	4	2	0	1	5	3	6

Table 1

In week 1 shoot *Footage 7*.

In week 2 shoots *Footage 4*.

In week 3 shoots *Footage 2*.

In week 4 shoots no *Footage*.

In week 5 shoots *Footage 1*.

…

In week 8 shoots *Footage 6*.

5.1.3 Examples of the Method

e.g.1:

i. Initial conditions

There are 3 movie stars

Star 1 is available in 1,2,3,5,6,7,8

Star 2 is available in 2,3,4,6,7,8

Star 3 is available in 1,4,5,6,7,

And 2 filming sites

Outdoor scene is available in 1,2,3,5,7,8

Indoor scene is available in 3,5,6,7,8

And one special prop

Special prop 1 is available in 3,5,7

And 7 *Footages*:

Footage 1 needs movie star 1, outdoor scene and special prop

Footage 2 needs movie star 2, indoor scene and special prop

Footage 3 needs movie star 1 and 2 and indoor scene

Footage 4 needs movie star 1 and 2 and outdoor scene

Footage 5 needs movie star 3 and indoor scene

Footage 6 needs movie star 1 and indoor scene

ii. *Constraint Matrices*

Movie star 1(*Constraint 1*) $A_{r_1} = (1,1,1,0,1,1,1,1)$

Movie star 2(*Constraint 2*) $A_{r_2} = (0,1,1,1,0,1,1,1)$

Movie star 3(*Constraint 3*) $A_{r_3} = (1,0,0,1,1,1,1,0)$

Outdoor scene (*Constraint 4*) $A_{r_4} = (1,1,1,0,1,0,1,1)$

Indoor scene (*Constraint 5*) $A_{r_5} = (0,0,1,0,1,1,1,1)$

Special prop 1(*Constraint 6*) $A_{r_6} = (0,0,1,0,1,0,1,0)$

iii. *Footage-Constraint Sets*

Footage 1: $F_1 = \{r_1, r_4, r_6\}$

Footage 2: $F_2 = \{r_2, r_5, r_6\}$

Footage 3: $F_3 = \{r_1, r_2, r_5\}$

Footage 4: $F_4 = \{r_1, r_2, r_4\}$

Footage 5: $F_5 = \{r_3, r_5\}$

Footage 6: $F_6 = \{r_1, r_5\}$

Footage 7: $F_7 = \{r_1, r_4\}$

iv. *Footage Matrices*

Footage 1: $B_1 = (0,0,1,0,1,0,1,0)$

Footage 2: $B_2 = (0,0,1,0,0,0,1,0)$

Footage 3: $B_3 = (0,0,1,0,0,1,1,1)$

Footage 4: $B_4 = (0,1,1,0,0,0,1,1)$

Footage 5: $B_5 = (0,0,0,0,1,1,1,0)$

Footage 6: $B_6 = (0,0,1,0,1,1,1,1)$

Footage 7: $B_7 = (1,1,1,0,1,0,1,1)$

v. *Filming Matrix*

$$C = \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} \begin{matrix} \to B_1 \\ \to B_2 \\ \to B_3 \\ \to B_4 \\ \to B_5 \\ \to B_6 \\ \to B_7 \end{matrix}$$

vi. *Schedule Matrix*

$E = (7,4,2,0,1,5,3,6)$

This step is crucial, and the selection of *Schedule Matrix* will be described later. It will be illustrated later and the enumeration method and weighting method will be used for comparison.

vii. Time table

According to the *Schedule Matrix*, a time table can be obtained, as shown in Table 2

Week	1	2	3	4	5	6	7	8
Footage No.	7	4	2	0	1	5	3	6

Table 2

So it takes 8 weeks to finish filming the complete movie.

5.1.4 Schedule Matrix

With the initial conditions, we can easily generated the *Filming Matrix*. Therefore, the key to our problem is how to generate the *Schedule Matrix* from the. *Filming Matrix*

For a *Filming Matrix*,

$$C = \begin{pmatrix} b_{11} & \dots & a_{1n} \\ \dots & \dots & \dots \\ a_{p1} & \dots & a_{pn} \end{pmatrix} \begin{matrix} \to B_1 \\ \to B_k \\ \to B_p \end{matrix}$$

Because there is only one 1(all *Footages* must be taken) on each row and at most one ‘1’ on each column (at most one *Footage* is planned to be taken every week), when ‘1’ once occurs in a row or column, we can eliminate the corresponding row and column.

To make the final schedule as short as possible, we may start with picking the element of ‘1’ in the first column and eliminate the corresponding row or column and then repeat this process until the last row is crossed out. Finally we can obtain a schedule.

i. Enumeration Method.

e.g.2:

The *Filming Matrix* with 7 weeks and 4 *Footages*

$$C=\begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ \cancel{1} & \cancel{1} & \cancel{0} & \cancel{0} & \cancel{1} & \cancel{0} & \cancel{0} \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}\begin{array}{l} \to B_1 \\ \to B_2 \\ \cancel{\to B_3} \\ \to B_4 \end{array}$$

The first '1'appears in the third and the forth row in first column, so we cross out the first column and the third row, and let $\boldsymbol{E}$= (3…

The matrix turns to:

$$C'=\begin{pmatrix} 0 & 1 & 0 & 1 & 1 & 1 \\ \cancel{1} & \cancel{0} & \cancel{1} & \cancel{0} & \cancel{1} & \cancel{1} \\ 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}\begin{array}{l} \to B_1 \\ \cancel{\to B_2} \\ \to B_4 \end{array}$$

Then we cross out the second row and the first column and let $\boldsymbol{E}$= (3,2…

The matrix turns to:

$$C''=\begin{pmatrix} \cancel{1} & \cancel{0} & \cancel{1} & \cancel{1} & \cancel{1} \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}\begin{array}{l} \cancel{\to B_1} \\ \to B_4 \end{array}$$

Then we cross out the first row and the first column and let $\boldsymbol{E}$= (3,2,1…

Here we cannot go any further. So we go back to the first column and try the second '1' (we have tried the first '1' before). We pick the fourth row in the first column

$$C=\begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ \cancel{1} & \cancel{0} & \cancel{1} & \cancel{0} & \cancel{0} & \cancel{0} & \cancel{0} \end{pmatrix}\begin{array}{l} \to B_1 \\ \to B_2 \\ \to B_3 \\ \cancel{\to B_4} \end{array}$$

In the same way, we obtain

$$C'=\begin{pmatrix} 0 & 1 & 0 & 1 & 1 & 1 \\ \cancel{1} & \cancel{0} & \cancel{1} & \cancel{0} & \cancel{1} & \cancel{1} \\ 1 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}\begin{array}{l} \to B_1 \\ \cancel{\to B_2} \\ \to B_3 \end{array}$$

$$C''=\begin{pmatrix} \cancel{1} & \cancel{0} & \cancel{1} & \cancel{1} & \cancel{1} \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}\begin{array}{l} \cancel{\to B_1} \\ \to B_3 \end{array}$$

And we get $\boldsymbol{E}$= (4,2,1,0,3)

If the *Schedule Matrix* is still not satisfying, we can keep going using enumeration method until the best answer $\boldsymbol{E}$= (4,3,1,2).is obtained.

ii. Weighting Method

If there are only a few *Constraints*, we can use the enumeration method till we get the best schedule. But when things get more complicated, the computation time will increase rapidly. Enumeration method may not help us to get the *Schedule Matrix* within a reasonable time even with the help of computers. Thus, we designed a weighting method which could help us get an approximate answer faster.

We observed that a *Footage* that can be shot only in a particular week is more restrictive than a *Footage* that can be shot in any week. Thus it is reasonable to give it a higher priority in the scheduling. So we define the weight for each *Footage* as the inverse to the availability of the *Footage*:

$$\lambda=\frac{\text{total number of weeks}}{\text{the number of '1's in a certain line}}$$

, and rearrange the *Filming Matrix* with higher-weighted *Footages* at top and lower ones at bottom. In this way we can efficiently get a satisfying result.

e.g.3:

Take the example from e.g.1:

$$C=\begin{pmatrix} 0&0&1&0&1&0&1&0\\ 0&0&1&0&0&0&1&0\\ 0&0&1&0&0&1&1&1\\ 0&1&1&0&0&0&1&1\\ 0&0&0&0&1&1&1&0\\ 0&0&1&0&1&1&1&1\\ 1&1&1&0&1&0&1&1 \end{pmatrix}\begin{matrix}\to B_1\\ \to B_2\\ \to B_3\\ \to B_4\\ \to B_5\\ \to B_6\\ \to B_7\end{matrix}$$

We can infer that

$$\lambda(B_1)=\frac{8}{3},\lambda(B_2)=\frac{8}{2},\lambda(B_3)=\frac{8}{4},\ \lambda(B_4)=\frac{8}{4},\lambda(B_5)=\frac{8}{3},\lambda(B_6)=\frac{8}{5},\lambda(B_7)=\frac{8}{6}$$

And thus $\lambda(B_2)>\lambda(B_1)=\lambda(B_5)>\lambda(B_3)=\lambda(B_4)>\lambda(B_6)>\lambda(B_7)$

$$\begin{array}{c} \begin{matrix}1&2&3&4&5&6&7&8&\text{(Week)}\end{matrix}\\ C(1)=\begin{pmatrix} 0&0&1&0&0&0&1&0\\ 0&0&1&0&1&0&1&0\\ 0&0&0&0&1&1&1&0\\ 0&0&1&0&0&1&1&1\\ 0&1&1&0&0&0&1&1\\ 0&0&1&0&1&1&1&1\\ 1&1&1&0&1&0&1&1 \end{pmatrix}\begin{matrix}\to B_2\\ \to B_1\\ \to B_5\\ \to B_3\\ \to B_4\\ \to B_6\\ \to B_7\end{matrix} \end{array} \to \begin{array}{c} \begin{matrix}1&2&4&5&6&7&8\end{matrix}\\ C(2)=\begin{pmatrix} 0&0&0&1&0&1&0\\ 0&0&0&1&1&1&0\\ 0&0&0&0&1&1&1\\ 0&1&0&0&0&1&1\\ 0&0&0&1&1&1&1\\ 1&1&0&1&0&1&1 \end{pmatrix}\begin{matrix}\to B_1\\ \to B_5\\ \to B_3\\ \to B_4\\ \to B_6\\ \to B_7\end{matrix} \end{array}$$

$$\to \ldots \to C(6)=\begin{pmatrix} 0&0&1\\ 1&0&1 \end{pmatrix}\to C(7)=\begin{pmatrix}1&0\end{pmatrix}$$

Respectively we can get that $e_3=2,e_5=1,\ldots,e_1=7$, then $E=(7,4,2,0,1,5,3,6)$

The result may not be optimal, but this method is more efficient than the enumeration method with answer close to optimal. What's more, this method can be easily implemented with the computer simulation.

5.2 Question 2

In Question 1, if one of the *Constraints* is changed (maybe caused by the injury of stars, the changing of the weather on the outdoor scene, and the damage of some special props.), the result will be affected to various extents. Now we are going to analyze how our model adjusts the schedule with such changes.

5.2.1 Analyzing Accidents and Adjusting

i. A movie star may get injured in an accident, which will cause him/her to rest for a period of time.

ii. Weather may get worse, such as possibility of raining rising, which will lower the availability of outdoor scenes.

iii. Special props may be damaged, which means replacement cannot be found immediately.

Accidents mentioned above affect further filming. We find that if we input the new *Constraints* to the model we established in Question 1 as new initial conditions, we will have the subsequent plan. (Actually it just generate a new schedule)

Rationality Explanation: Our model can quickly generate a reasonable schedule according to the actual situation.

For example: The Original *Filming Matrix* is the same as that in e.g.2

$$C=\left(\begin{array}{ccccccc} \cancel{1} & \cancel{0} & \cancel{1} & \cancel{0} & \cancel{0} & \cancel{0} & \cancel{0} \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 \end{array}\right)\begin{array}{l} \cancel{\rightarrow B_4} \\ \rightarrow B_3 \\ \rightarrow B_1 \\ \rightarrow B_2 \end{array}$$

$E=(4,3,1,2)$,

Assume when the first week of shooting was done, and an actor was caught in an accident, making *Footage 1* and *Footage 3* (both require the actor) impossible to be filmed in the next two weeks. We can list the new initial conditions.

$$C'=\left(\begin{array}{cccccc} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \end{array}\right)\begin{array}{l} \rightarrow B_3 \\ \rightarrow B_1 \\ \rightarrow B_2 \end{array}$$

Put these into the model and we know $E'=(2,0,0,3,1)$

The schedule is showed in Table 3 below (The plan of the first week is the same as before)

Week	1	2	3	4	5	6
Footage No.	4	2	0	0	3	1

Table 3

Originally it takes 4 weeks to complete the filming and now it's postponed to 6 weeks, as $E=(4,3,1,2)$ turn to $E_2=(4,2,0,0,3,1)$. Here, our model has given the solution.

5.3 Question 3

Since we consider movie stars, filming sites and props as the same type of *Constraints*, we can then define an importance index for a certain *Constraint*

$$\text{importance index}=\frac{\dfrac{\text{number of times the constraint is needed (in 20 footages)}}{\text{average filming time}}}{\text{the possibility the constraint can be filmed}}$$

5.3.1 Example

e.g. 4:

Assume there's a movie that must meet the following conditions:

i. 20 *Footages* in total.
ii. A Large *Footage* needs 5 elements(3 movie stars, 1 filming site, 1 special prop)
iii. A Medium *Footage* needs 3 elements(2 movie stars, 1 filming site)
iv. A Small *Footage* needs 2 elements(1 movie stars, 1 filming site)
v. The proportion of Large, Medium and Small *Footage* is 2: 5: 3 (aka 4 Large *Footages*, 10 Medium *Footages* and 6 Small *Footages*.)

Because of the differences of movies are mostly represented by the proportion of indoor and outdoor scenes, so we studied 4 different types of films listed below.

Constraints	Availability
y_1(stars)	0.9
y_2	0.8
y_3	0.8
z_1 (indoor scene)	0.8
z_2 (outdoor scene)	0.7
w_1(props)	0.8
w_2	0.4

Movie Types	Proportions of indoor scene and outdoor scene
1	9: 1
2	6: 4
3	4: 6
4	1: 9

Table 4

Through the formula given above we can get the table below:

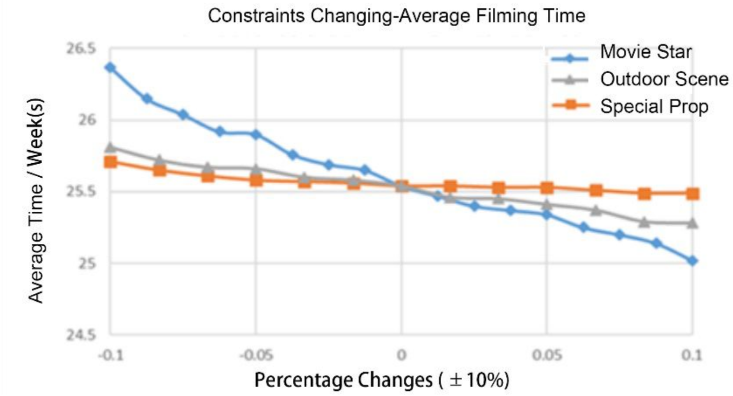
Constraints		y_1	y_2	y_3	z_1	z_2	w_1	w_2
importance index	Type 1	0.56	0.63	0.63	0.9	0.11	0.1	0.2
	Type 2	0.6	0.67	0.67	0.64	0.49	0.11	0.21
	Type 3	0.59	0.67	0.67	0.42	0.72	0.11	0.21
	Type 4	0.51	0.57	0.57	0.09	0.93	0.09	0.18

Table 5.

We found that indoor scenes is a highly stable *Constraint*, so we took only movie stars, film scenes (z_1 excluded) and special props into further consideration.

We observed that when the proportion of indoor scene and outdoor scene is relatively high, the importance index for movie stars is the highest. And when the proportion of indoor scene and outdoor scene is relatively low, on the other hand, the importance index of outdoor scene will become the highest.

In order to verify our conclusion, we used type 2 as an example, we take y_3 (movie star), z_2 (outdoor scene [weather]), w_2 (special prop) (all with the highest importance index separately) as basis, and drew the following graph.



Graph 1

It's obvious that change in the availability of movie stars causes the biggest schedule altering compared to other *Constraints*. In order to further prove this point, we analyzed the range of these set of data.

We get:

Range (movie stars) =1.35, Range (special props) =0.22, Range (weather) =0.53

So the *Constraint* with the highest importance index affects the average filming time the most.

5.3.2 Conclusions

No matter it is a movie star, filming site or special prop, the *Constraint* with the highest importance index is the key factor, which varies depending on the different type of movies.

6 Analysis of the Model

6.1 Simulation in Real Situations

Through a program we wrote in Visual Basic (using weighting method, source code see in the Appendix), we can get a schedule that meets your requirements as long as you can provide a detailed list of availability of each *Constraint* including movie stars, filming sites, special props and the *Constraints* needed for every *Footage*.

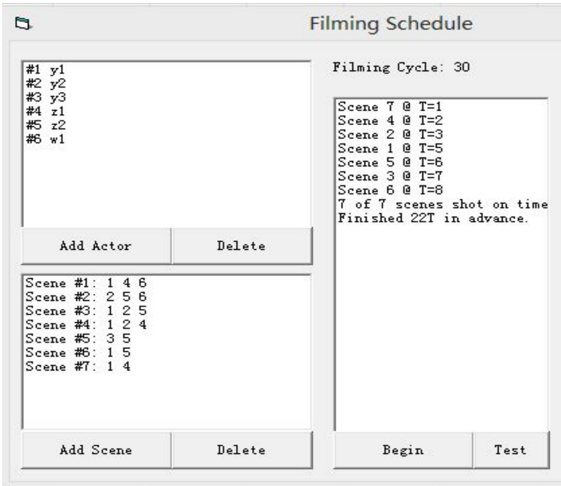
P.S.: Considering that when the movie first begin filming, the setting up of indoor scene may not have been completed yet, so indoor scene is always unavailable in the first week.

Take e.g. 1 for example: Input these *Constraints* into the computer and we will get the data in Picture 1.

Constraint No.	*Constraints*	Available times
1	y_1	1-3,5-8
2	y_2	2-4,6-8
3	y_3	1,4-7
4	z_1(outdoor scene)	1-3,5,7-8
5	z_2(indoor scene)	3,5-8
6	w_1	3,5,7

Footage No.	Requires
1	1,4,6
2	2,5,6
3	1,2,5
4	1,2,4
5	3,5
6	1,5
7	1,4

Table 6



Picture 1

We can extract this table below

Week	1	2	3	4	5	6	7	8
Footage NO.	7	4	2	0	1	5	3	6

Table 7

It takes at least 8 weeks to finish shooting.

So we have proven that the program is reliable and accurate.

6.2 **Random Simulation**

6.2.1 Simulation

In order to further validate the model's reliability and stability, we use the computer to conduct random simulations, in order to determine the average filming period of a movie. To reduce the amount of calculation, we assume there are 3 movie stars, 2 film sites(indoor scene and outdoor scene),2 special props.(Because the number of movie stars in each film won't be large, and film scenes are mainly indoor scenes and outdoor scenes, and the special prop usage is low)

In order to make the model more realistic, we use e.g. 4 in the simulation

i. 20 *Footages* in total.
ii. A Large *Footage* needs 5 elements(3 movie stars, 1 film scene, 1 special prop)
iii. A Medium *Footage* needs 3 elements(2 movie stars, 1 film scene)
iv. A Small *Footage* needs 2 elements(1 movie stars, 1 film scene)
v. The proportion of Large *Footages*, Medium *Footages* and Small *Footages* is 2: 5: 3(4 Large *Footages*, 10 Medium *Footages* and 6 Small *Footages*.)

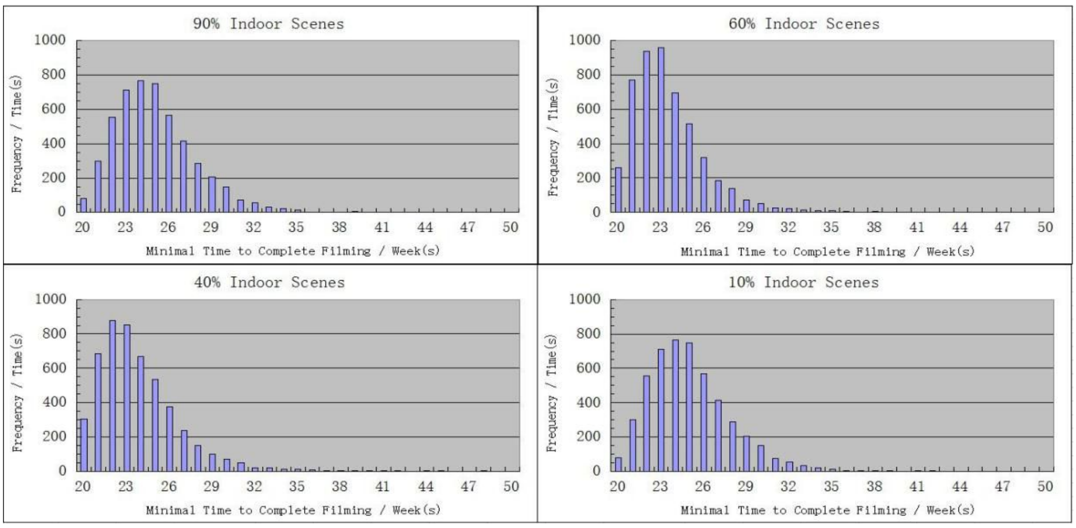
Because proportions of indoor scene and outdoor scene among different types of films, we studied these four different types of films.

Constraints	Availability
y_1	0.9
y_2	0.8
y_3	0.8
z_1(indoor scene)	0.8
z_2(outdoor scene)	0.7
w_1	0.8
w_2	0.4

Movie Types	the proportions of indoor and outdoor scene
1	9: 1
2	6: 4
3	4: 6
4	1: 9

Table 8

Then we can get the following graph (Each type simulated 5000 times)



Graph 2

	Type 1 90% indoor scene	Type 2 60% indoor scene	Type 3 40% indoor scene	Type 4 10%indoor scene
Median	25	23	23	27
Mode	24	23	22	25
Average	25.01	23.53	23.81	27.58
Variance	8.32	6.77	8.53	16.77
Standard deviation	2.88	2.60	2.92	4.09
frequency of samples falling in the σ interval	0.6416	0.8397	0.7989	0.6902
frequency of samples falling in the 2σ interval	0.9572	0.9563	0.9572	0.9578

Table 9

6.2.2 Conclusions

As for the findings in the chart, several points stand out

i. Frequency show a skewed normal distribution to average time, while the mean, median and mode are almost the same.
ii. For movies with a balanced ratio of indoor and outdoor scenes, results provided by this model are mostly below 25 weeks, the σ interval covers around 80% of the data and the variance is below 3.
iii. For movies with extreme amount of indoor or outdoor scenes, results provided by this model are still below 28 weeks, the σ interval reaching 65% coverage of the data and the variance is around 4.

Analysis: When the movie is full of one kind of scene (indoor or outdoor), it’s common for different *Footages* to conflict with each other, lengthening the average filming cycle. So the increment in avg. Filming cycle is reasonable.

Again, using tables and graphs, we have proved the efficiency and the stability of our model, with performance in extreme conditions also matching the reality.

7 Advantages, Disadvantages and Improvements

7.1 Advantages

i. Simpleness: This model is easy to use, you get immediate results as soon as you input all the *Constraints*.
ii. Accurateness: It can provide you with an accurate schedule.
iii. Reliability: Result provided by this model will not conflict with real situation.
iv. Operability: Using matrices, all computing became executable.

v. Flexibility: This model can adjust its output in time according to the reality.

vi. Stability: Results of the model do not vary greatly from each other.

vii. This model uses the concept of weight and is represented as the ability to recognize more important scenes and arrange them prior to others.

viii. This model considers movie stars, filming sites and special props as one single type of *Constraints*.

7.2 Disadvantages

i. To reduce the amount of calculation, we set the unit time as a week. Actually, the available time of movie stars, filming sites and the special props may be calculated by the days.

ii. To reduce the amount of calculation, what we get is only a relatively good solution, not always the best one.

7.3 Improvements

i. Setting the unit time as a day, or using subroutines for more detailed arrangements in each week will make the model more practical.

ii. When processing the *Filming Matrix*, if the weighting and rearranging process is repeated every time after the crossing-out of the matrix, then the output would become slightly better.

8 Reference

【1】 线性代数与解析几何[M] 廉庆荣主编 高等教育出版社 2008

【2】 数学模型(第四版)[M]高等教育出版社 姜启源 2011

【3】 高等数学(第六版)[M]同济大学应用数学系编著 高等教育出版社 2007

【4】 Visual Basic 6.0 中文版编程方法详解[M]李鸿吉编著 科学出版社 2002

【5】 中学数学建模与赛题集锦(第二版)[M]上海市中学生数学知识应用竞赛组织委员会编著 复旦大学出版社 2014

【6】 数学建模方法及其应用(第二版)[M] 韩中庚编著 高等教育出版社 2009

【7】 创意制片完全手册: 从项目策划到营销发行 [M]Maureen A. Ryan 编著 马瑞青译 2015

【8】 中级运筹学[M] 胡知能主编 科学出版社 2008

【9】 电影制作日程安排表 1, [EB/OL], http://wenku.baidu.com/link?url=S1k2i-LR7aIEfuqc2KcAOZcFjcg0uwUgvDZ6cd_n3IjVNqS7bE3FLxfW4lBoYfCppLy4CGASVwwjh7rSV46TN0F_h_5KBVrE5vd9T4DWpDO, xrb515437431 , 2011-11-05

【10】 电影拍摄日程, [EB/OL], http://wenku.baidu.com/link?url=Z6c-m8QO4-owUrfZXzykoOwSzBc9quST86thD7suBqPgv8yKy990sbKzmZUnHa4ywUnh4exHaXO0ddlkVPPmIHvAOOlYoBJEwMgYXJLiRVW, 西瓜太瓜, 2014-04-05

9 Appendix

```
Private actorcount, scenecount, tmin, tmax
Private actor() As Long, scene() As Long
Private A() As Integer, myschedule() As Integer
Private ac() As Integer, seq() As Integer
Private temp() As Integer
Private mycycle
Private testcount, testpass, tavg
Private outputstr As String
Private Sub addactor_Click()
    On Error GoTo error1:
    actorcount = actorcount + 1
    mytag = InputBox("Tag this resource?", , "Resource" & actorcount)
    ReDim Preserve actor(mycycle, actorcount) As Long
    Do
        intervalstart = Int(Val(InputBox(mytag & " is available from ..." & Chr(10) & "Enter nothing
to finish input.")))
        If intervalstart = 0 Then Exit Do
        intervalend = Int(Val(InputBox(mytag & " is available till ...")))
        If intervalstart > intervalend Then GoTo error1
        For t = intervalstart To intervalend
            actor(t, actorcount) = 1
        Next t
    Loop
    listactor.AddItem "#" & actorcount & " " & mytag
    Exit Sub
error1:
    MsgBox "Error Encountered. Your data will not be saved."
    actorcount = actorcount - 1
    ReDim Preserve actor(mycycle, actorcount) As Long
    Exit Sub
End Sub
Private Sub addscene_Click()
    On Error Resume Next
    actorstr = ""
    ReDim Preserve scene(1, scenecount + 1) As Long
    scenecount = scenecount + 1
    Do
        actornum = Val(InputBox("This scene requires the presense of actor#..." & Chr(10) & "(enter
ONE at one time, 0 to confirm)"))
        If actornum > 0 Then
            scene(0, scenecount) = scene(0, scenecount) + 2 ^ actornum
            actorstr = actorstr & " " & actornum
        End If
    Loop Until actornum <= 0
    listscene.AddItem "Scene #" & scenecount & ": " & actorstr
End Sub
Private Sub cmdbegin_Click()
    On Error GoTo error1
    'Initialization
    Erase ac: Erase seq: useds = "": myshot = 0
    ReDim A(1 To scenecount, 0 To mycycle) As Integer, myschedule(1 To mycycle) As Integer
    ReDim temp(1 To scenecount, 1 To mycycle) As Integer
    ReDim ac(0 To scenecount) As Integer
    ReDim seq(0 To scenecount) As Integer
    Dim conflict As Integer
    If listscene.ListCount = 0 Then GoTo noscene
    'build a 2D grid
    For myscene = 1 To scenecount
```

```
        For t = 1 To mycycle
            actorsum = scene(0, myscene)
            ta = 1
            If ta = 1 Then
                For i = actorcount To 1 Step -1
                    If actorsum \ 2 ^ i = 1 Then
                        actorsum = actorsum - 2 ^ i
                        ta = ta * actor(t, i)
                        If ta = 0 Then A(myscene, t) = 0: Exit For
                        If actorsum = 0 And ta = 1 Then A(myscene, t) = 1: temp(myscene, t) = 1:
ac(myscene) = ac(myscene) + 1: Exit For
                    End If
                Next i
            End If
        Next t
    Next myscene
    'arrange them in the sequence of avilability reversed order
    For i = 1 To scenecount
        For j = 1 To scenecount
            On Error Resume Next
            If ac(i) < ac(seq(j)) Or ac(seq(j)) = 0 Then
                For k = scenecount To j + 1 Step -1
                    seq(k) = seq(k - 1)
                Next k
                seq(j) = i: Exit For
            End If
        Next j
    Next i
    For myseq = 1 To scenecount
        For i = 1 To mycycle
            A(myseq, i) = temp(seq(myseq), i)
            A(myseq, 0) = seq(myseq)
        Next i
    Next myseq
    'get final arrangements
    For t = 1 To mycycle
        For s = 1 To scenecount
            conflict = 0
            For i = 1 To mycycle
                If myschedule(i) = A(s, 0) Then conflict = 1: Exit For
            Next i
            If A(s, t) = 1 And conflict = 0 Then myschedule(t) = A(s, 0): myshot = myshot + 1: Exit
For
        Next s
    Next t
    listresult.Clear
    For i = 1 To mycycle
        If myschedule(i) > 0 Then listresult.AddItem "Scene " & myschedule(i) & " @ T=" & i: tlast
= i
    Next i
    listresult.AddItem myshot & " of " & scenecount & " scenes shot on time."
    If scenecount - myshot = 0 Then
        listresult.AddItem "Finished " & mycycle - tlast & "T in advance."
        testpass = testpass + 1
        tavg = tavg + tlast
        outputstr = outputstr & Chr(10) & tlast
    Else: listresult.AddItem "Failed to finish on time."
    End If
    Exit Sub
error1:
```

```
    MsgBox "ERROR."
    Exit Sub
noscene:
    MsgBox "Generate a SCENE first."
    Exit Sub
End Sub
Private Sub cmdtest_Click()
'Initialize
    Dim testav(1 To 9) As Single
    For i = 1 To 9
        testav(i) = testset(i).Text
    Next i
    mycycle = testset(10).Text
    testcount = 0: testpass = 0: tavg = 0: outputstr = ""
'Repeating Tests
For myrepeat = 1 To InputBox("Repeat test for...", , 5000)
    testcount = testcount + 1
    Randomize
    lblcycle.Caption = "Filming Cycle: " & mycycle
    Erase actor: Erase scene
    ReDim actor(mycycle, 7) As Long
    ReDim scene(1, 20) As Long
    listactor.Clear
    listscene.Clear
    listresult.Clear
    actorcount = 7
    scenecount = 20
    For i = 1 To 7
        listactor.AddItem "Test Data"
        For t = 1 To mycycle
            If Rnd <= testav(i) Then actor(t, i) = 1
        Next t
    Next i
    actor(1, 4) = 0 'first day indoor is not avilable
    For s = 1 To 20
        listscene.AddItem "Test Data"
        'decide the size of scene
        If s = 1 Then stype = 1
        If s = 5 Then stype = 2
        If s = 15 Then stype = 3
        'decide indoor/outdoor set
        If Rnd <= testav(8) Then nset = 4 Else nset = 5
        Select Case stype
        Case 1
            If Rnd <= testav(9) Then nprop = 6 Else nprop = 7
            scene(0, s) = 14 + 2 ^ nset + 2 ^ nprop
        Case 2
            Rnd
            Select Case Rnd
            Case Is <= 0.33
                nact1 = 1
                nact2 = 2
            Case Is <= 0.67
                nact1 = 1
                nact2 = 3
            Case Else
                nact1 = 2
                nact2 = 3
            End Select
            scene(0, s) = 2 ^ nact1 + 2 ^ nact2 + 2 ^ nset
```

```
        Case 3
            Rnd
            Select Case Rnd
            Case Is <= 0.33
                nact1 = 1
            Case Is <= 0.67
                nact1 = 2
            Case Else
                nact1 = 3
            End Select
            scene(0, s) = 2 ^ nact1 + 2 ^ nset
        End Select
    Next s
    Call cmdbegin_Click
    DoEvents
Next myrepeat
    MsgBox "Tested " & testcount & " times in total. " & testpass & "(" & Format(testpass / testcount,
"0.00%") & ") passed." & Chr(10) & "Avg Time Elapse: T=" & tavg / testpass
    Clipboard.Clear
    Clipboard.SetText outputstr
End Sub
Private Sub delactor_Click()
    On Error GoTo error1
    ReDim Preserve actor(mycycle, actorcount - 1) As Long
    actorcount = actorcount - 1
    listactor.RemoveItem (listactor.ListCount - 1)
    Exit Sub
error1:
    MsgBox "No Item to Delete."
    Exit Sub
End Sub
Private Sub delscene_Click()
    On Error GoTo error1
    ReDim Preserve scene(1, scenecount - 1) As Long
    scenecount = scenecount - 1
    listscene.RemoveItem (listscene.ListCount - 1)
    Exit Sub
error1:
    MsgBox "No Item to Delete."
    Exit Sub
End Sub
Private Sub Form_Load()
    tmin = 9999
    mycycle = Int(Val(InputBox("Your estimated filming cycle?", , 30)))
    lblcycle.Caption = "Filming Cycle: " & mycycle
End Sub
Private Sub lblcycle_DblClick()
    myconfirm = MsgBox("Changing your filming cycle will erase all data inputed. " & Chr(10) &
"Continue anyway?", vbYesNo)
    If myconfirm = vbYes Then
        Erase actor: Erase scene: Erase A: Erase temp
        listactor.Clear
        listscene.Clear
        mycycle = Int(Val(InputBox("Your estimated filming cycle?", , 30)))
        lblcycle.Caption = "Filming Cycle: " & mycycle
    End If
End Sub
```