Team Control Number: #6770

Problem Chosen: A

In order to best organize a triathlon with a 1500m swim, a 40K bike, and a 10K run, we calculated the average time in seconds it took for different divisions of participants based on age, gender, and skill level to complete each segment of a triathlon using data from a previous event. By compiling this data, we were able to develop a model that minimized congestion and had road blockage time below 5.5 hours using incremented waves of participants with similar completion times.

The first part of our process was to divide the participants into different divisions in order to group and split them into various wave starts. All skill levels were grouped into two divisions based on gender, with further age classification for the large “Open” registrators. Using the times given by the Mayor to complete a previous triathlon of the same length, we calculated the average time in seconds it took for each of the 24 divisions to complete the swimming, cycling, and running segments, along with the two transition times in between. Then, the data was used to split large divisions and group smaller divisions with similar completion times together in order to create waves with a consistent number of contestants.

The second part of our process was to develop a model in order to appropriately sequence and separate wave starts with the goal of minimizing congestion as much as possible. Using a distance-time graph, we used the Q1, mean, and Q3 values of the average completion times for the participants in each wave start to create functions in order to illustrate the estimated range (Q1-Q3) of speeds that contestants in a certain wave start would have.

Lastly, we developed a refined model of the wave starts during the triathlon by using a logarithmic function to represent the “slowing down” of the participant during the swimming segment, a linear function to represent the constant speed of the contestant during the bike, and an exponential function to show the increase in speed of runners in the last segment, accommodating for the transitions during the race. To evaluate the effectiveness of our refined model, we proposed a cost function to calculate the cost of congestion and cost of time. Our goal is to reduce the cost as much as possible. In this cost function, time is considered least expensive as long as it fits within the 5.5 hours and congestions are usually more expensive because we think it is more important for participants to have a better experience with less congestions. To calculate the total cost, we used a combination of programming and integrals as demonstrated in 3.2. It is estimated that all participants will finish in 5.3 hours, successfully within the time limit. In conclusion, our proposed model was able to both minimize congestion, and keep the road blockage time below 5.5 hours.

Letter to the Mayor

Dear Mayor,

The Triathlon event consists of 1500 meter swim, 40 kilometer cycling, and 10 kilometer running, sponsored by the Super Tread Race Company, will be a tremendous opportunity for an approximately 2000 athletes and contestants of all levels: from professionals to amateurs.

In order to minimize the time period of roads closures during the event and minimize the congestion time to make an enjoyable environment for all athletes, we utilized "division" and "wave starts," which enables participants to race with others who have similar race pace and to start in small time intervals. We decided to send the fastest athletes (Wave 1 - pro and premier athletes) first and let the slowest wave race the last so that the rapid ones would perform their best without any disturbances from slower people. For instance, Wave 2 would be sent 150 seconds after Wave 1 to ensure the time intervals in between the each wave, which helps to minimize congestions.

Through the event, the organisation, we, incorporated awards in each division to motivate all contestants to be competitive and enjoy the event. The division is divided into 4 levels in terms of past Triathlon experiences and abilities: Pros, Premiers, Open, and CLY and ATH. Awarding in different divisions will attract professional and experienced athletes to participate the event. It will also let new or older participants, who are often discouraged by faster professionals, to have a chance to receive an award even though they are not top contestants overall; however, they might be top athletes for their division. Providing various awards will encourage all participants from diverse backgrounds and experiences.

At the same time, we accomplished to hold the Triathlon under 5.5 hours: our plan only takes 5.3 hours, which is 12 minutes less than the regulation. Keeping the road closure time under 5.5 hours will also allow local residents, businesses, and traffic not to get hindered by the event. In addition to the time length of the road closure, according to the race schedule, the first Triathlon begins at 7:30AM; many athletes will stay in the hotel for a night before the race to fully prepare for the event. This will bring profits to the local businesses, both hotel industry and shops. Our plan not only considers the well-being and satisfaction of the participants but also takes the interests of the local government and people.

Attached on the next page is the schedule of the Triathlon event. We express gratitude for your trust in us to plan this significant event, which we hope will help prosper the local area and will grow into a world-class event with many high-class athletes competing.

Best,

Group 6770

The Triathlon Race Schedule

Time	Wave starts/ Functions	Time	
5:30 A.M.	Athletes race packet pick-ups	8:32:30	Wave 26: Open Female 50-54 & Open Male 18-24
7:30:00 A.M.	Wave 1: all Pros and Premiers T1 transition areas open Local roads are closed	8:35:00	Wave 27: Open Female 30-34-1
7:32:30	Wave 2: Open Male 30-34-1	8:37:30	Wave 28: Open Female 30-34-2
7:35:00	Wave 3: Open Male 30-34-2	8:40:00	Wave 29: Open Female 30-34-3
7:37:30	Wave 4: Open Male 30-34-3	8:42:30	Wave 30: Open Female 25-29-1
7:40:00	Wave 5: Open Male 30-34-	8:45:00	Wave 31: Open Female 25-29-2 T2 transition areas open
7:42:30	Wave 6: Open Male 40-44-1	8:47:30	Wave 32: Open Male 60+
7:45:00	Wave 7: Open Male 40-44-2	8:50:00	Wave 33: Open Female 45-49
7:47:30	Wave 8: Open Male 40-44-3	8:52:30	Wave 34: Open Female 35-39-1
7:50:00	Wave 9: Open Male 40-44-4	8:55:00	Wave 35: Open Female 35-39-2
7:52:30	Wave 10: Open Male 25-29-1	8:57:30	Wave 36: Open Female 40-44 & Open Female 18-24-1
7:55:00	Wave 11: Open Male 25-29-2	9:00:00 A.M.	Wave 37: Open Female 40-44 & Open Female 18-24-2
7:57:30	Wave 12: Open Male 25-29-3	9:02:30	Wave 38: CLY & Open Female 55-59
8:00:00 A.M.	Wave 13: Open Male 45-49-1	9:05:00	Wave 39: Open Female 60 & ATH
8:02:30	Wave 14: Open Male 45-49-2	9:50	T1 transition areas are closed
8:05:00	Wave 15: Open Male 45-49-3	10:30	T2 transition areas are closed
8:07:30	Wave 16: Open Male 35-39-1	12:50 P.M.	Estimated end of race
8:10:00	Wave 17: Open Male 35-39-2	1:00 P.M.	Local roads open
8:12:30	Wave 18: Open Male 35-39-3		
8:15:00	Wave 19: Open Male 35-39-4		
8:17:30	Wave 20: Open Male 35-39-5		
8:20:00	Wave 21: Open Male 50-54-1		
8:22:30	Wave 22: Open Male 50-54-2		
8:25:00	Wave 23: Open Male 50-54-3		
8:27:30	Wave 24: Open Male 55-59-1		
8:30:00	Wave 25: Open Male 55-59-2		

Table of Contents:

1. The Model

1.1 Background

A triathlon is a sequence of three different athletic events, generally long distance swimming, cycling and running. In order to attract various athletes to compete, divisions must be carefully chosen so that people of various physical conditions would be given an opportunity to win an award. The athletes are ranked by their course completion time, which is the sum of the completion times of each segment as well as the transition time between events. Since the sponsor, the Super Tread Race Company, wants to ensure the fairness of this triathlon so that it would prosper as a world-class event that attracts high-class athletes, it is necessary that congestion has minimum effect on the contestants' records. Minimizing the total time spent on the event is also necessary, as the Mayor does not want the local roads to remain blocked because of the event for an extended period of time. Our aim is to organize a triathlon with an estimate of 2000 runners in a 1500 meter swim, 40 kilometer cycling, and 10 kilometer running that takes place in a time under 5.5 hours, finding a balance between minimizing congestion and minimizing time by the use of "wave starts," or sending off groups of contestants in a set order with small time intervals. Possible adjustments to the race distances or other conditions to improve congestion will then be investigated.

We were provided with a data table of a recent triathlon with a 1500 meter swim, 40 kilometer cycling, and 10 kilometer running consisting of 3217 total runners by the Mayor as a source to refer to while organizing our triathlon.

1.2 Restatement of the Problem

The challenges in organizing this triathlon are: creating appropriate divisions to attract various athletes of different gender, age, and skill level, grouping waves with an adequate number of people, determining the order of waves in order to avoid as much congestion as possible, determining the time intervals between waves, and calculating the amount of congestion and its cost. The effect of change in the total distance on the amount of congestion will also have to be investigated to explore the advantages we may achieve in future events.

2. Definitions

2.1 Assumptions

- Only people 18 years old and above can register for this race
- To standardize our calculations, we changed the expression of time from HH:MM:SS into seconds by 3600 * HH + 60 * MM + SS
- Each wave roughly consists 50 people, and no more than 75 peoples
- The race is starting at 10 AM.

2.2 Defining the Variables

Q1: the middle value in the first half of the rank-ordered data set (25th percentile)
Q2: the median value in the set (50th percentile)
Q3: the middle value in the first half of the rank-ordered data set (75th percentile)
IQR: a measure of statistical dispersion, being equal to the difference between 75th and 25th percentiles, or between upper and lower quartiles
Y1: the previous wave
Y2: the current wave

3. The Models

3.1 The Crude Model

3.1.1 Divisions

In a triathlon competition, it is conventional to divide the contestants into different divisions based on their specific age group, gender and their skill level for the fairness of awards. In our race, there are three categories of skill level: "Pro", "Premier", and "Open." All skill level categories were split into two divisions based on gender: male and female, in order to ensure fairness for the racers. Due to the large estimated number of contestants for the "Open" category, we further divided them into nine age groups: 18-24, 25-29, 30-34, 35-39, 40-44, 45-49, 50-54, 55-59, and 60+. Lastly, two special divisions: Clydesdale and Athena, were added for men over 220 pounds and women over 165 pounds respectively.

Below is a table of all 24 divisions:

Table 1: List of Divisions

Clydesdale	Athena	Male Pro	Male Premier	Female Pro	Female Premier
Open Male 18-24	Open Male 25-29	Open Male 30-34	Open Male 35-39	Open Male 40-44	Open Male 45-49
Open Male 50-54	Open Male 55-59	Open Male 60+	Open Female 18-24	Open Female 25-29	Open Female 30-34
Open Female 35-39	Open Female 40-44	Open Female 45-49	Open Female 50-54	Open Female 55-59	Open Female 60+

To assist us in organizing wave start times, we ranked the divisions using a scale from 1-24, 1 having the average fastest times, 2 having the next fastest times, and so on. In order to estimate the performance of racers in each division based on speed, we employed the data set given by the Mayor. Using programming (see code 1 attached), we were able to calculate the average total race time for each division. Our estimations were made more accurate by eliminating outliers from the average in each division: any time below Q1 - 1.5IQR or above Q3 + 1.5IQR.

In addition, we used ratios to predict the number of contestants in each division, organizing the contestants of the previous triathlon into our divisions. Since we are expecting roughly 2000 people for our event, we scaled down the 3217 participants to 2000 people:

number of participants expected = number of participants in the dataset * $\frac{2000}{3217}$.

Below are our results:

Table 2: Ranking the Divisions Based On Data Set of the Recent Race

Division	Name of the Division	Avg. Time to Finish Race (excluding outliers) unit:seconds	# of People In Previous Triathlon (3217 Total)	# of People Estimated for This Triathlon (2000 Total)
1	Male Pro	6886.143	7	4
2	Female Pro	7407.8333	6	4
3	Male Premier	7842	49	30
4	Female Premier	8629.1667	18	11
5	Open Male 30-34	10355.17429	362	225
6	Open Male 40-44	10357.02865	356	221
7	Open Male 25-29	10465.89669	246	153
8	Open Male 45-49	10467.9058	283	176
9	Open Male 35-39	10517.12834	385	239
10	Open Male 50-54	10673.2654	220	137
11	Open Male 55-59	10973.96	155	96
12	Open Female 50-54	11045.804	76	47
13	Open Male 18-24	11147.41667	36	22
14	Open Female 30-34	11522.114	209	130
15	Open Female 25-29	11538.85821	134	83
16	Open Male 60+	11558.13592	104	65
17	Open Female 45-49	11632.767	118	73

18	Open Female 35-39	11706.697	145	90
19	Open Female 40-44	11709.465	132	82
20	Open Female 18-24	11798.6087	23	14
21	CLY	11805.31667	60	37
22	Open Female 55-59	12081	37	23
23	Open Female 60+	12586.625	24	15
24	ATH	13376.77419	32	20

3.1.2 Waves

Since we define each wave to have roughly 50-75 participants, it is necessary for us to split large divisions and group smaller divisions together, bearing the idea in mind that people with similar skill levels (similar average time to complete the race) should race together. For example, we grouped Male Pro, Female Pro, and Male Premier participants together because of their similar skill levels and their relatively small numbers of participants, and, we divided the Open Male 30-34 division into 4 groups, each group with roughly 50 contestants.

Below are our 39 waves:

Wave 1 : Male Pro, Male Premier, Female Pro, Female Premier
Wave 2-5: Open Male 30-34
Wave 6-9: Open Male 40-44
Wave 10-12: Open Male 25-29
Wave 13-15: Open Male 45-49
Wave 16-20: Open Male 35-39
Wave 21-23: Open Male 50-54
Wave 24, 25: Open Male 55-59
Wave 26: Open Female 50-54 and Open Male 18-24
Wave 27-29: Open Female 30-34
Wave 30, 31: Open Female 25-29
Wave 32: Open Male 60+

Wave 33: Open Female 45-49

Wave 34, 35: Open Female 35-39

Wave 36, 37: Open Female 40-44 and Open Female 18-24

Wave 38: Clydesdale and Open Female 55-59

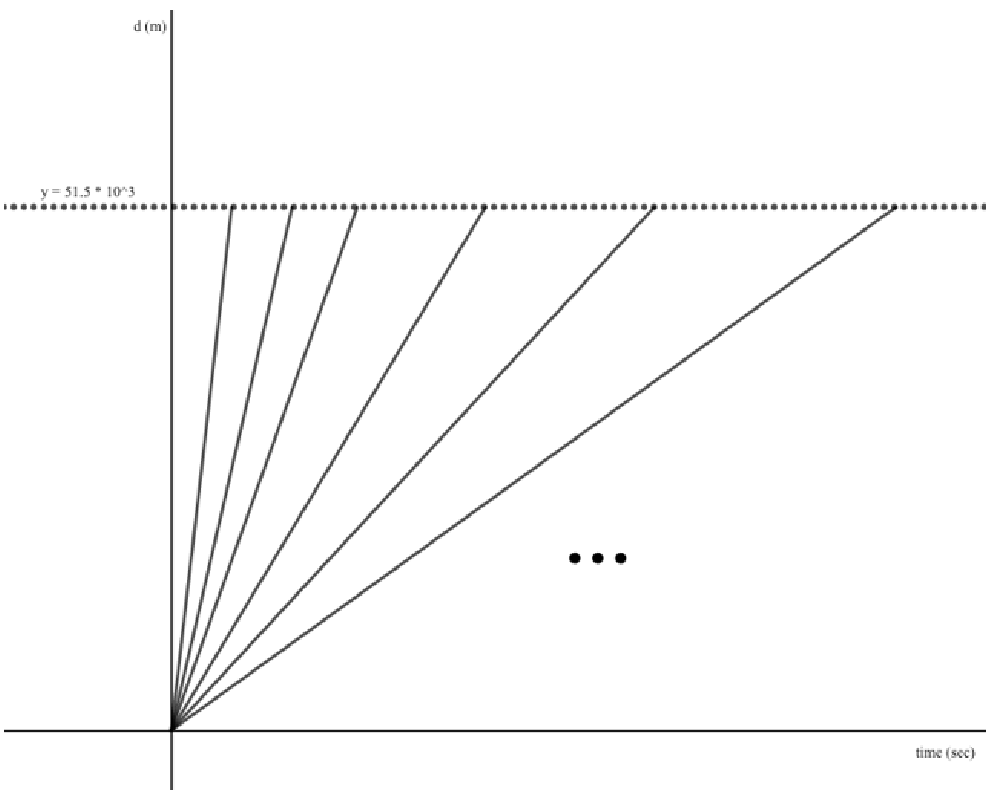
Wave 39: Open Female 60+ and Athena

3.1.2 Order of Waves

To minimize the congestion within the time span of 5.5 hours, we originally proposed a schedule with waves that have participants from the fastest divisions going first and waves with the slowest divisions going last.

Below is the graph of our idealized model:

Graph 1



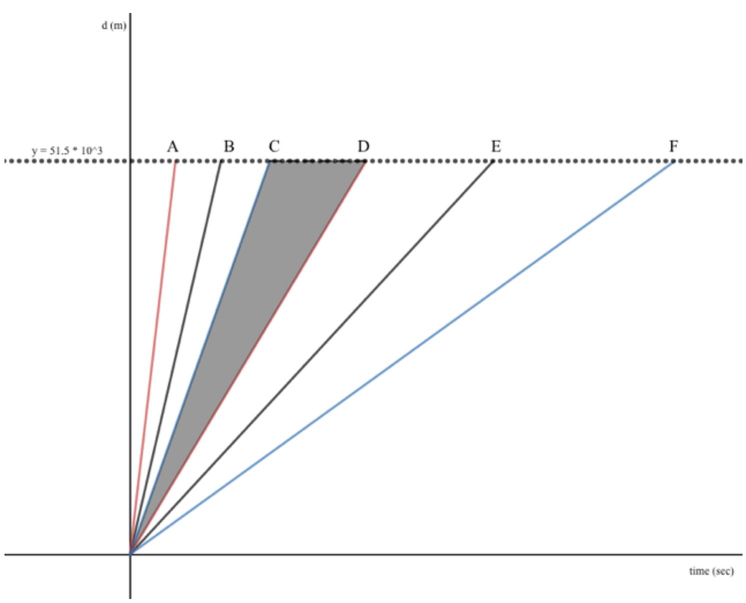
In this distance-time graph, the x-axis represents time (in seconds), and the y-axis represents distance (in meters). In this idealized model, we originally assumed that the participants in the triathlon would be moving in a constant velocity throughout the whole event, resulting in a linear distance-time graph. The waves start the race almost all at the same time, in the sequence of the wave with the fastest divisions to the wave with the slowest division. Ideally, the groups wouldn't have any congestions as the slower waves would never be able to catch up with the waves in front of them. Wave 1 would arrive at the ending point first, and wave 39 would arrive last. The total time required for the race would simply be the time required for the last wave, wave 39, to finish the race, which is roughly 13376.774 sec on average.

However, this model is not realistic for the following two major reasons:

1) In reality, we cannot physically start all waves at the same time or almost at the same time, as the spacing capabilities will have a limit to how many participants can be ready to start racing in a given period of time. Time spacings between each wave must be considered in order to plan a real event.
2) Although we are only considering the average speed of each wave, implying that all runners in a wave would move at a given speed, realistically, the waves would consist of participants with a range of various speed, causing huge problems of congestion.

Below is a demonstration of the congestion problem of releasing all waves almost at the same time:

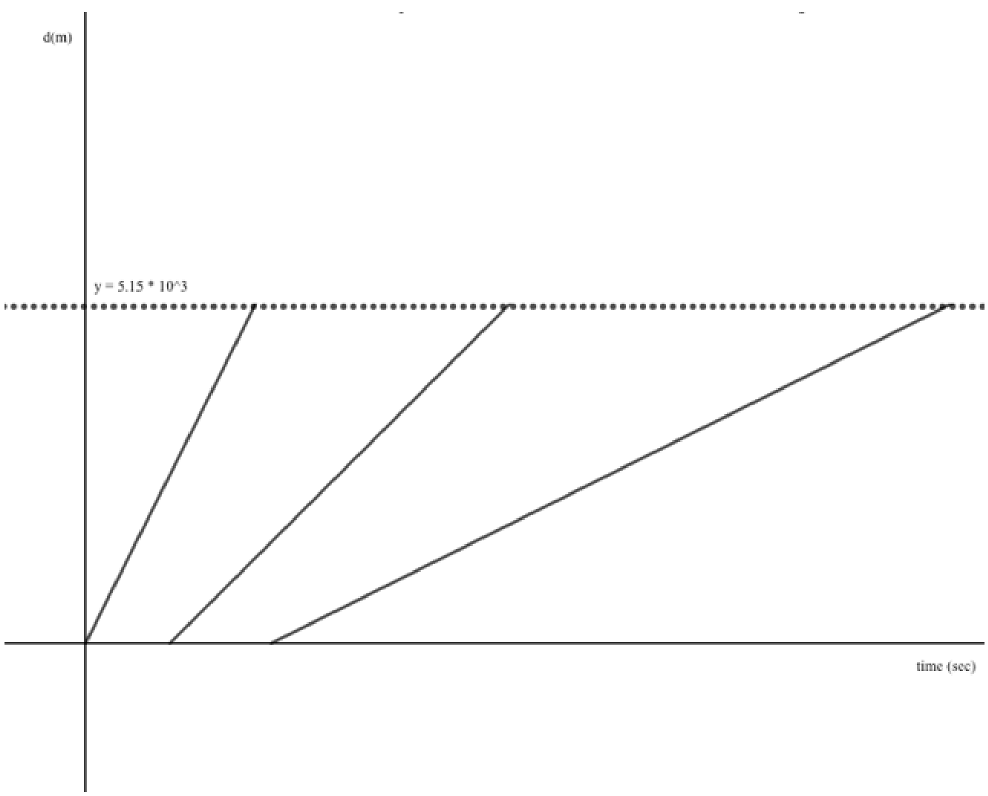
Graph 2



In this graph, you can imagine that the origin is O. Line OB and OE are the average speeds of two consecutive waves, while the red and blue lines represent the bounds in which most participants' speed in that wave will fall under. While OB is the average of the faster wave, which we shall call wave X, OE is the average of the slower wave, which we shall call wave Y. In reality, wave X consists of runners with a whole spectrum of speed, which is reflected in the area of ΔAOD. Similarly, wave B covers the area of ΔCOF. A huge congestion would appear in the overlapping area of ΔAOD and ΔCOF, which is ΔCOD. For every consecutive wave, there would going to be a congestion like ΔCOD, suggesting the impracticality of this ideal graph.

To minimize the shaded region (congestion), we needed to increase the time interval between each wave, resulting in the increase of the total time to solve the problem of congestion. By shifting the slower waves to the right, the shaded region of ΔCOD would be minimized.
Below is an improved model:

Graph 3

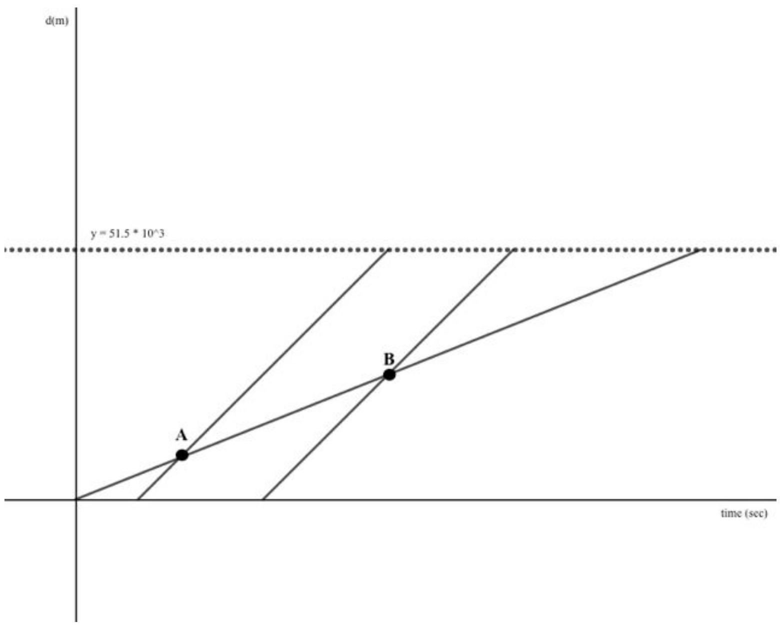


In Graph 3, the three waves represented are released in longer intervals, decreasing the congestion as desired.

Now raise the question: is it possible for us to change the sequence of the waves? What would happen to our congestion and road block time?

The following is a demonstration of the consequences of the changes of the sequence of the waves:

Graph 4



In this distance-time graph, where slower waves are going to be released before faster waves, faster waves are going to run into participants of the slower waves, causing congestions in points A and B. In this case, changing the sequence of the waves increases the congestion dramatically and cannot save time, another manifestation that graph 3 model is the best possible model in terms of sequence of waves.

***3.1.3 Spacing of Wave*s**

As what we discussed in section 3.1.2, we increased the time intervals to decrease the congestion within the time limit of 5.5 hours, meaning the last wave cannot be released later than:

5.5 hours - the time the last wave needs to finish the race = 19800s - 13376.774s = 6423.226 seconds into the race

As there are 39 waves in total, there are 38 time intervals. Each time interval should be roughly 6423.226 seconds / 38 = 169.032 sec, which is close to 2.5 minutes. Therefore, we define our time intervals between each wave to be roughly 150 seconds.

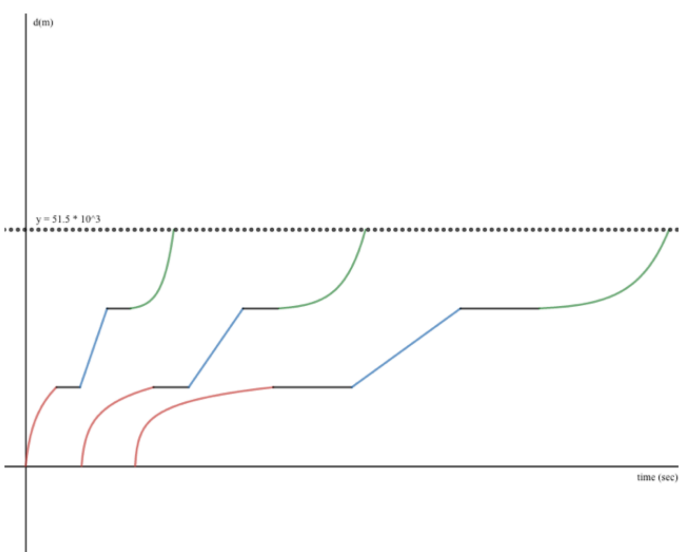
3.2 Refined Model

3.2.1 Modeling Best Fit

In order to create a more refined model for our distance-time graph of the contestants, we used different nonlinear functions for different events. A logarithmic function was used to demonstrate the "slowing down" of the swimmers during that segment, a linear function was used to demonstrate the relatively consistent speed of the cyclists, and an exponential function was used to demonstrate the increase of velocity of the runners as they gain momentum. A horizontal line was inserted in between each segment to indicate the transition time (T1 and T2) where there is no change in distance over the change of time.

As a result, we used the following model to represent the distance-time graph of all three events:

Graph 5



In the more refined model above, the red logarithmic segment represents the swimming event, the blue represents cycling, the green represents running, and the black horizontal lines between each event represent the transitions. The different average slopes of the graphs represent the different skill levels of the contestants of each wave.

3.2.2 Evaluating the Cost

To evaluate the effectiveness of our refined model, we propose a cost function to calculate the congestion of our model.

∑ Total Cost = ∑ of individuals cost = ∑ travel time cost + ∑ delay/early cost + ∑ congestion cost in race

To calculate the delay/early cost and the congestion cost in the race, we considered all of the participants inside of the bounds of Q1 to Q3. The congestion or overlaps happens when the Q1 of one wave runs into the Q3 of the wave before it.

Table 3: Definitions of Cost of Congestion and Cost of Time on the scale of $0 to $1

Delay/Early Cost	$0.3
1 Cycling Congestion Area	$0.8
1 Running Congestion Area	$0.6
1 Swimming Congestion Area	$1
1000s	$1

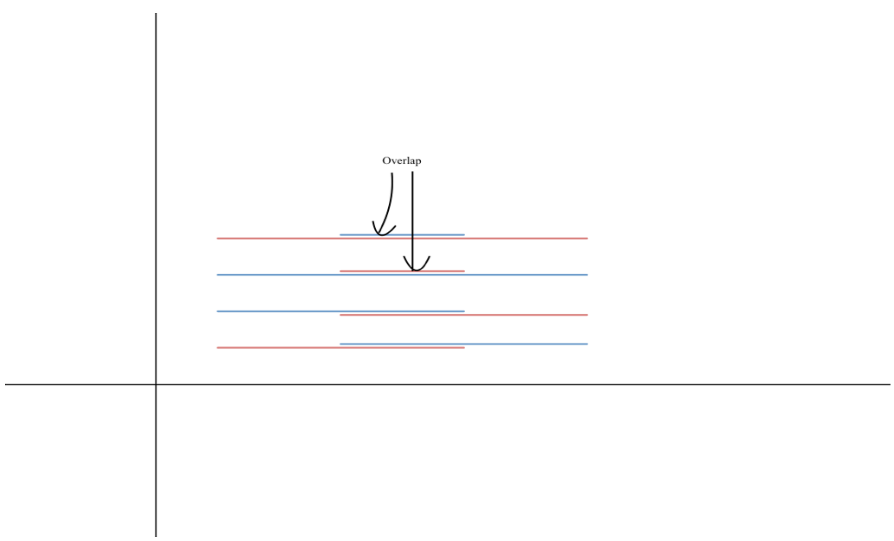
The values of the delay/early costs, the congestion costs and the time costs were assigned with respect to their importance. The costs for the congestion that occurs during swimming are assigned the maximum cost, $1 per 1 area, because the time records of swimmers that swim in assigned lanes will be most heavily affected by running into swimmers of the previous waves. The congestion costs for cycling was also given a relatively high value of $0.8 per unit area as the contestants would be accompanied by their bicycles, making it more difficult to avoid other contestants during the race. The congestion costs during running is assigned $0.6 per unit area, as the contestants move in a relatively slower speed than in rest of the segments, making it easier to avoid chaos. The costs for congestion in transition periods are assigned the smallest value, as none of the contestants would be changing their distance. We assigned $1 per 1000s as the time cost, giving it relatively the smallest value, as the triathlon is a competition where the ranks of the runners should be determined fairly, giving more importance to minimizing congestion that minimizing the total time.

In our model:

a) ∑ of travel time cost = Total Road Block Time * cost = 5.5 hours * cost = 19800s * $1 / 1000 seconds = $1.98
b) ∑ of delay/early Cost :

Delay or Early Cost happens when one wave arrives too early or late for their preparation phase(either t1 or t2) such that its preparation phase overlaps with that of the previous wave. The graph below shows 4 situations of overlapping:

Graph 6



Distance-time graph

Red: transition section of the Q3 graph of wave y1, Blue: transition section of the Q1 graph of wave y2

Because it is inefficient to find out the occurrences of overlapping by hand, we created a program (see attached code 2).

Below is the pseudo-code that we used to determine whether overlapping occurs and, if it does, what the length of overlapping is:

Pseudo-Code 1

```
Void overlaps(int y1, int y2)
{
      int b1 = data[y1][2];  //beginning of y1
      int e1 = data[y1][2] + data[y1][4]; //end of y1
      int b2 = 150 + data[y2][1]; //beginning of y2
      int e2 = 150 + data[y2][1] + data[y2][3]; //end of y2
  if (e1<b2 || e2<b1)
        Print "No there's not an overlap" ;
  Else {
         Print "Yes there is an overlap";
          int len; //calculate the overlapping len
          if (b1<=b2 && b2<=e1 && e1<=e2)    len = e1 - b2;
          else if (b2<=b1 && b1<=e2 && e2<=e1)  len = e2-b1;
          else if (b1<=b2 && b2<=e2 && e2<=e1)  len = e2 - b2;
          else   len = e1 - b1;
          Print len;  //printing the overlapping len
  }
```

We applied this code to both T1 and T2.

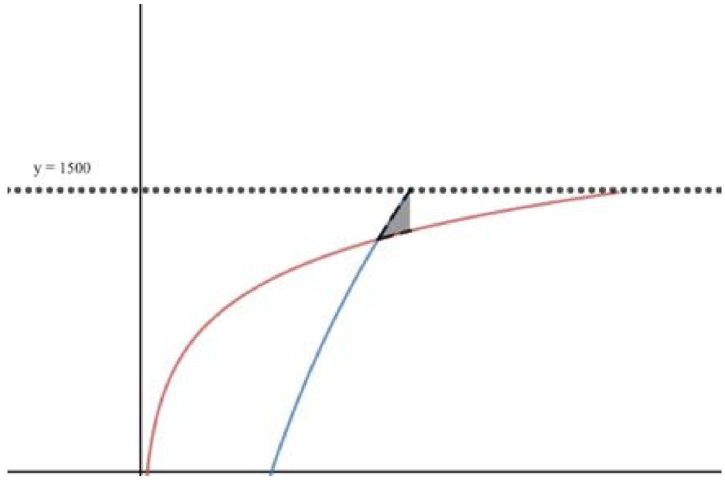
Below is what we found out about the overlapping:

$\sum$ Early/Delay Cost = $\sum$ length of overlapping * cost = \$.3 * length = 10013 * \$.3 = \$3003.9

c) Cost of Congestion during the race

Congestion occurs when Q1 graph of the current wave runs into the Q3 graph of the previous wave during races.

Graph 7 of possible congestion area in swimming race:



Distance(m)-time(s) graph

Red: swimming section of Q3 of the previous wave

Blue: swimming section of Q1 of the current wave

X: time(s)

Y: Distance(m)

Below is the pseudo-code that we used to judge if there is going to be occurrences of congestion area in the swimming race:

Pseudo-Code 2:

```
Void overlaps(y1,y2)   {   //y1 is previous wave, y2 is current wave
     if (time of end of y2 swimming< time of end of y1 swimming)
          Print "Yes there is congestion"
      else
          Print "No there is no congestion"
}
```

If there was congestion, we calculated the shaded area.

Let the red function be called $f(x) = k1 * e^{x}$, and the blue function be called $g(x) = k2 * e^{x-150}$

Set $f(x) = g(x)$.

Then, $k1 * e^{x} = g(x) = k2 * e^{x-150}$

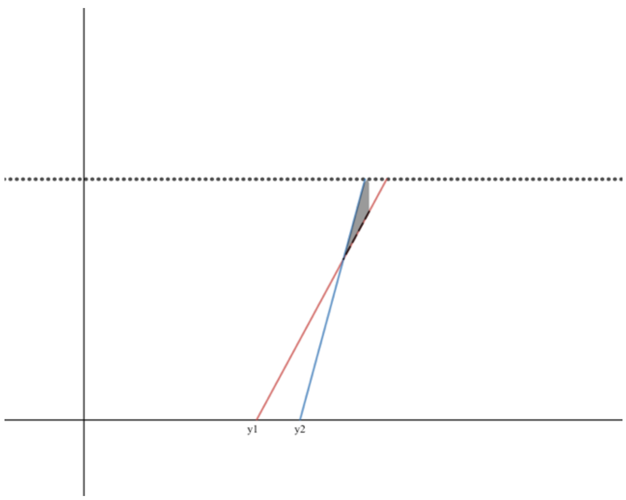
Solve for the root, and we get $x = t0$.

The shaded area = $\int_{a}^{b} 1500 - f(x)\,dx$

(b = time of end of y1 swimming, a = time of end of y2 swimming)

∑ Cost of congestion during the swimming race = summation of shaded area * 1$

Graph of possible congestion area in the cycling segment:



Distance(m)-Time(s) Graph

Red: swimming section of Q3 of the previous wave

Blue: swimming section of Q1 of the current wave

X: time(s)

Y: Distance(m)

Below is the pseudo-code that we used to judge if there is going to be occurrences of congestion area in the cycling segment:

Pseudo-Code 3:

```
Void overlaps(y1,y2)    {    //y1 is previous wave, y2 is current wave
    if (time of end of y2 cycling< time of end of y1 cycling && beginning of y2 cycling > beginning of y1 cycling )
                Print "Yes there is congestion"
         else
```

Print "No there is no congestion"

}

If there was congestion, we calculated the shaded area.

Let the red function be called f(x) = k1 * x + a, and the blue function be called g(x) = k2* x + b.

(a,b are constants). Set f(x) = g(x).

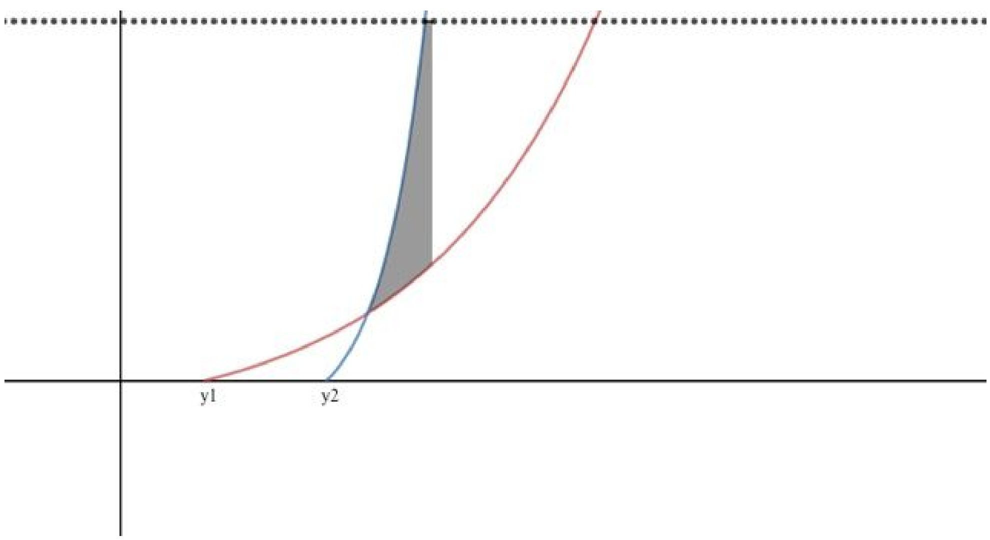
Then, k1 * x + a = g(x) =k2* x + b

Solve for the root, and we get x = t0.

The shaded area = $\int_a^b g(x)-f(x)\,dx$ (b = time of end of y1 cycling, a = time of end of y2 cycling)

∑ Cost of congestion during the cycling race = ∑ shaded area * $0.8

Graph 8 of possible congestion area in running race:



Distance(m)-Time(s) Graph

Red: running section of Q3 of the previous wave

Blue: running section of Q1 of the current wave

X: time(s)

Y: Distance(m)

Below is the pseudo-code that we used to judge if there is going to be occurrences of congestion area in the running race:

Pseudo-Code 4:

```
Void overlaps(y1,y2)    {    //y1 is previous wave, y2 is current wave
    if (time of end of y2 running< time of end of y1 running && beginning of y2 running > beginning of y1
running)
            Print "Yes there is congestion"
        else
            Print "No there is no congestion"
}
```

Let the red function be called $f(x) = k1 * x + a$, and the blue function be called $g(x) = k2 * x + b$. (a,b are constants). Set $f(x) = g(x)$.

Then, $k1 * x + a = g(x) = k2 * x + b$

Solve for the root, and we get $x = t0$.

$$\text{The shaded area} = \int_{a}^{b} g(x) - f(x)\, dx$$

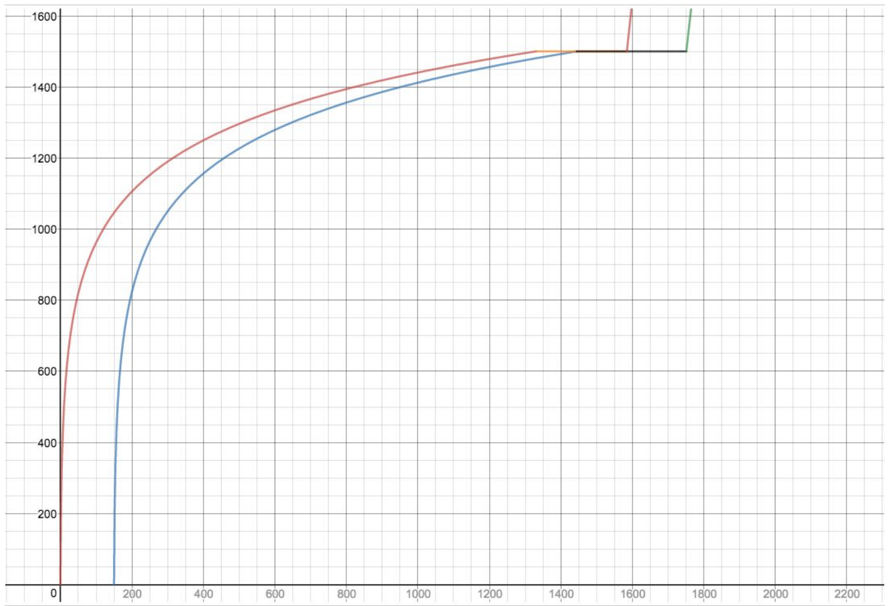
(b = time of end of y1 cycling, a = time of end of y2 cycling)

$\sum$ Cost of congestion during the running race = $\sum$ shaded area * \$0.6

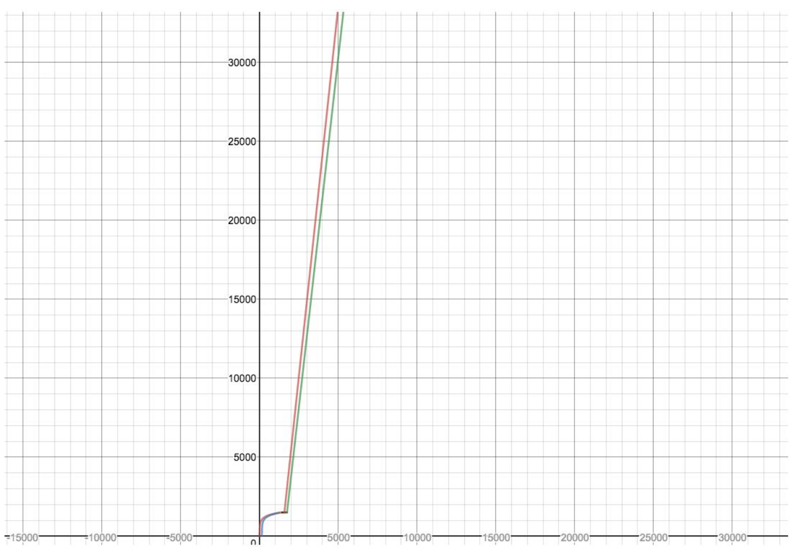
3.2.3 Numerical Example of the Costs

Sample Calculations for Congestions between Wave 1 and Wave 2:

Graph 9 for swimming and t1 segments for Wave 1(Q3) and Wave 2(Q1):



Graph 10 for cycling segments for Wave 1(Q3) and Wave 2(Q1):



Graph 11 for running & t2 segments for Wave 1(Q3) and Wave 2(Q1):

Wave 1 is Male Pro, Male Premier, Female Pro, and Female Premier. Wave 2 is Male 30-34. The sample calculation consists of Q3 of Wave 1 (referred as Q3), the 25th percentile of Wave 1, and Q1 of Wave 2 (referred as Q1), the 75th percentile of Wave 2, to determine the existence of congestions between the two waves. A congestion would be defined by an intersection between the functions of Q3 and Q1. We defined the x-coordinate of the two functions as the time they take to complete each event, and y-coordinate as the distance covered by the runners.

The logarithmic function that models the distance-time graph of the swimmers is $y = 208.503\, ln\,(x + 1)\ \{0 \leq x \leq\ 1330.625\}$ for Q3 and $y\ =\ 209.291\ ln\,(x\ -\ 149)\ \{150 \leq x \leq\ 1445\}$ for Q1. The basic form of the function is $y\ =\ k\ ln\,(x + 1)$ when k is the slope; we made horizontal shift by 1 since we want the function to begin at the origin and ln(1) becomes 0. There is a horizontal shift of 149 to make the function (150, 0). We acquire the slope of the functions through plugging in one of the (x,y) values of the function. For Q3, we used (1330.625, 1500) to calculate: $k\ ln\,(1330.625\ + 1)\ =\ 1500,\ k\ =\ 208.503.$ For Q1, we used (1445, 1500) to calculate: $k\ ln\,(1445\ -\ 149)\ =\ 1500,\ k\ =\ 209.291.$ Starting from (0,0), Q3 will reach (1330.625, 1500) after swimming event. 1330.625 s can be acquired through taking an average of the four groups consisting Wave 1: $\frac{1360 + 1069.5 + 1321 + 1571}{4}$ s. 1500m indicates the distance of the swimming event. After 150 s, Q1 will depart and finish the swimming event at (1445, 1500). 1445 s can be obtained through 1295 + 150, adding up the starting time and the time it took for Q1 to swim. There would be no congestion in the swimming event since both of the graphs do not intersect during the interval of (0, 1500) s.

After the swimming event, Q3 will finish T_1 at (1584.75, 1500) while Q1 will finish at (1751, 1500). There will be an intersection in the two functions (Q3 and Q1), which is y = 1500 since Q1 starts while Q3 is on T_1, which slows down the transition and creates a congestion as a result. In order to find the overlapping time period in T_1, we subtracted the finish time of Q3 in T_1 (1584.75) by the start time of Q1 in T_1 (1445): 139.75 seconds.

The linear function for cycling is y = 9.384x - 13371.294 $\{1584.75 \leq x \leq 5847.125\}$ for Q3 and y = 8.861x - 14015.611 $\{1751 \leq x \leq 6265\}$ for Q1. The slope for Q3 (9.384) can be calculated through $\frac{\Delta y}{\Delta x} = \frac{41500 - 1500}{5847.125 - 1584.75}$ and the slope of Q1 (8.861) can be calculated in the same way: $\frac{41500 - 1500}{6265 - 1751}$. The y-intercept for the functions can be calculated through plug in one of the values; For Q3, we plugged in (1584.75, 1500) to get the value of y-intercept: 9.384(1584.75) + b = 1500, b = -13371.294. Using same process, for Q1, we plugged in (1751, 1500) to calculate the y-intercept: 8.861(1751) + b = 1500, b = -14015.611. Q3 completes the cycling event at (5847.125, 41500) and Q1 finishes at (6265, 41500). 41500 m represents the distance of the cycling race and the finishing times can be obtained by adding up the time Q3 or Q1 finished T_1 and the time they need to bike. Therefore, 1584.75 + 4262.375 = 5847.125 s for Q3 and 1751 + 4514 = 6265 s for Q1. There would be no congestion in the cycling event since the two linear functions do not meet in the domain of [1500, 41500]. Q3 will finish T_2 at (5919.875, 41500) while Q1 will finish at (6376, 41500). We got 5919.875 s through adding up the finishing time of the cycling event (5847.125 s) and the average time the four groups in the Wave 1 took to complete T_2 ($\frac{54 + 88 + 64 + 85}{4} =$ 72.75 s). Similar to Q3, 6376 s for Q1 can be acquired through the addition of the finishing time of the cycling event and the time Q1 took to complete T_2 (6265 + 111 = 6376 s). There would be no congestion in T_2, which can be described as y = 41500 as a function, because Q3 finishes T_2 before Q1 starts T_2.

There is a congestion in between Q3 and Q1 while running. The basis of the running functions are exponential: $y = ke^{bx} + c$.

The function for Q3 is $y = 4e^{(0.00150x-5)} + 41307$ $\{5919.875 \leq x \leq 8354.375\}$ and the function for Q1 is $y = 3.333e^{(0.00155x-4.8)} + 40963$ $\{6376 \leq x \leq 9376\}$. Q3 starts to run at (5831.38, 41500) and ends running at (8305.88, 51500). Q1 starts to run at (6395.03, 41500) and ends running at (9395.03, 51500). Since Q1 begins to run when Q3 is still running, there is a congestion, which indicates that the graph of Q1 and Q3 will meet during the run event. Mathematically proven, furthermore, the two functions have an intersection of (6859.215, 42099.44).

∑ cost = ∑ travel time cost + ∑ delay/early cost + ∑ congestion cost in race

= 8354 sec * $1/10000 sec + 139.75 * $.3 + 17.454 * $.6

= $.8 + $41.925 + $10.472

= $53.197

3.3 Application of Models: Answering the Questions

3.3.1 Part I Problems

1. Refer to Table 1 for the divisions that will be used in our Triathlon.

We set the starting time of our triathlon to be 7:30 A.M., which is a conventional start time of a race, and also to avoid making the contestants compete in the early afternoon, which is the warmest hours of the day. As explained in 3.1.3, we have set the time intervals between waves to be 150s.

2. Below is the schedule of our wave start times.

Table 4: Wave Start Times

Key: if there is going to be multiple waves for one division, we are going to attach "- group number" to the name of the division. (E.g. Open Male 30-34-1)

Time	Wave
7:30:00 A.M.	Wave 1: all Pros and Premiers
7:32:30	Wave 2: Open Male 30-34-1
7:35:00	Wave 3: Open Male 30-34-2
7:37:30	Wave 4: Open Male 30-34-3
7:40:00	Wave 5: Open Male 30-34-4
7:42:30	Wave 6: Open Male 40-44-1
7:45:00	Wave 7: Open Male 40-44-2
7:47:30	Wave 8: Open Male 40-44-3
7:50:00	Wave 9: Open Male 40-44-4

7:52:30	Wave 10: Open Male 25-29-1
7:55:00	Wave 11: Open Male 25-29-2
7:57:30	Wave 12: Open Male 25-29-3
8:00:00 A.M.	Wave 13: Open Male 45-49-1
8:02:30	Wave 14: Open Male 45-49-2
8:05:00	Wave 15: Open Male 45-49-3
8:07:30	Wave 16: Open Male 35-39-1
8:10:00	Wave 17: Open Male 35-39-2
8:12:30	Wave 18: Open Male 35-39-3
8:15:00	Wave 19: Open Male 35-39-4
8:17:30	Wave 20: Open Male 35-39-5
8:20:00	Wave 21: Open Male 50-54-1
8:22:30	Wave 22: Open Male 50-54-2
8:25:00	Wave 23: Open Male 50-54-3
8:27:30	Wave 24: Open Male 55-59-1
8:30:00	Wave 25: Open Male 55-59-2
8:32:30	Wave 26: Open Female 50-54 & Open Male 18-24
8:35:00	Wave 27: Open Female 30-34-1
8:37:30	Wave 28: Open Female 30-34-2
8:40:00	Wave 29: Open Female 30-34-3
8:42:30	Wave 30: Open Female 25-29-1
8:45:00	Wave 31: Open Female 25-29-2
8:47:30	Wave 32: Open Male 60+
8:50:00	Wave 33: Open Female 45-49
8:52:30	Wave 34: Open Female 35-39-1

8:55:00	Wave 35: Open Female 35-39-2
8:57:30	Wave 36: Open Female 40-44 & Open Female 18-24-1
9:00:00 A.M.	Wave 37: Open Female 40-44 & Open Female 18-24-2
9:02:30	Wave 38: CLY & Open Female 55-59
9:05:00	Wave 39: Open Female 60 & ATH

3.3.2 Part II Problems

Adjusting the race distances in future triathlons can bring us several advantages in terms of congestion and road closure time. For instance, when we reduces the reduce the race distances, the total time for the race required is going to decrease, and thus the road-block time would decrease as well. However, in this case, the congestion would be still be the same if not larger since time intervals haven't been changed. If we reduce the race distance and we increase the time interval, we are going to have less congestion, while we would have to block the road for longer time.

In order to further explore the advantages we will achieve in terms of congestion and road closure time by adjusting the race distances in future triathlons, we simulated an occasion where we decrease the total race distance by ⅓. We programmed a code (see attached) to determine how many times congestion, or the intersection between two waves, happen during the whole triathlon. Originally, congestion happened 110 times--when we decreased the distance by ⅓, thereby decreasing each record time by ⅓, and increased the time intervals 3-folds, while increasing the total time to 5.99 hours (108.88% increase), we decreased the number of congestion to 46 times, which is a decrease to 41.8% of the original amount of congestion. Therefore it is calculated that changing the race distance more affects the amount of congestion than the change of total time. By adjusting the race distance and increasing the time intervals in adequate increments, we will have an advantage of organizing a more fair event with less congestion to affect the time records of the contestants that costs a similar or even less total time.

Another advantages of reducing the race distance is that this event can probably attract more participants in the future since it is less physically tiring to do the triathlon, even though more participants could possibly mean more road blocking time. However, GDP loss due to the road locking can be compensated by the consumptions of the participants, and the reputation of the town can be sufficiently boosted.

4. Conclusion

To find a best mathematical model for Triathlon, we started out by organizing and assigning contestants to each division with the consideration of the fairness of the competition based on age, gender, and skill levels. Then we either split or group division together, allowing each wave to have roughly 50-75 people. After a discussion of the sequence of the waves to avoid congestion, we proposed a crude model with fastest wave going first and slowest wave going the last. To make our crude model more realistic, we applied ln function, linear function, exponential function respectively to the race of swimming, cycling, and running.

To further understand the effectiveness of our model, we proposed a cost function. Our goal is to reduce the cost as much as possible. In this cost function, time is considered least expensive as long as it fits within the 5.5 hours and congestions are usually more expensive because we think it is more important for participants to have a better experience with less congestions. To calculate the total cost, we used a combination of programming and integrals as demonstrated in 3.2.

The most notable strength of our model is that we solved the problem of congestion fairly well. We ensured most people would have a good experience with no much congestion by spreading the time interval between each waves. However, this method also renders us our most notable weakness that is we could not solve the problem of time well. Even though we managed to reduce our road blocking time to under 5.5 hours, we couldn’t minimize the time as much as we hoped, especially when time is considered relatively cheap in our cost function.

5. Appendix

5.1 Data Tables

A. Calculating the Average Time (in seconds) for All Divisions To Complete the Whole Race

	# of People	# of People (2000)	Mean	Q1	Q3	IQR	Q1 - 1.5 * IQR	Q3 + 1.5 * IQR	Outliers	Avg. Without Outliers
CLY TOTAL	60	37	11805.31667	10349	13105	2756	6215	17239	N/A	11805.31667
ATH TOTAL	32	20	13558.375	12343.5	14818.5	2475	8631	18531	19188	13376.77419
Male Pro TOTAL	7	4	6886.142857	6694	7016.5	322.5	6210.25	7500.25	N/A	6886.142857
Male Premier TOTAL	49	30	7842	7431	8208.5	777.5	6264.75	9374.75	N/A	7842
Female Pro TOTAL	6	4	7407.8333	7106	7537	431	6459.5	8183.5	N/A	7407.8333
Female Premier TOTAL	18	11	8629.1667	8279	8963	684	7253	9989	N/A	8629.1667
Male 18-24	36	22	11147.41667	9707	12368.5	2661.5	5714.75	16360.75	N/A	11147.41667
Male 25-29	246	153	10552.91057	9351	11420	2069	6247.5	14523.5	14793, 14983, 16405, 17088	10465.89669
Male 30-34	362	225	10518.00829	9429	11293	1864	6633	14089	14122, 14641, 14678, 14733, 15016, 15036, 15424, 15532, 15557, 16017, 16096, 16356	10355.17429
Male 35-39	385	239	10651.45455	9629	11372	1743	7014.5	13986.5	14100, 14474, 14492, 14903, 15258, 15288, 15289, 15423, 15948, 16099, 16130	10517.12834
Male 40-44	356	221	10444.03371	9512.5	11203.5	1691	6976	13740	14218, 14501, 14533, 14739, 14816, 15081, 15585	10357.02865
Male 45-49	283	176	10590.45936	9495	11528	2033	6445.5	14577.5	14665, 14887, 15043, 15534, 15712, 15746, 16371	10467.9058
Male 50-54	220	137	10868.50455	9857	11550.5	1693.5	7316.75	14090.75	14372, 14462, 14585, 14596, 14690, 15483, 16101, 17304, 17419	10673.2654
Male 55-59	155	96	11156.16774	9952	11983	2031	6905.5	15029.5	15033, 15232, 15279, 15574, 15776, 16447	10973.96
Male 60+	104	65	11621.31731	10137.5	12963	2825.5	5899.25	17201.25	18129	11558.13592
Female 18-24	23	14	11798.6087	10472.5	12697	2224.5	7135.75	16033.75	N/A	11798.6087
Female 25-29	134	83	11538.85821	10385	12645	2260	6995	16035	N/A	11538.85821
Female 30-34	209	130	11725.83254	10422.5	12458	2035.5	7369.25	15511.25	16059, 16160, 16889, 16929, 18296, 18766, 20133	11522.114
Female 35-39	145	90	1514.724756	10421	13066	2645	6453.5	17033.5	17508, 19602, 21468	11706.697
Female 40-44	132	82	11846.17424	10462	12725	2263	7067.5	16119.5	16988, 17123, 19063	11709.465
Female 45-49	118	73	11744.02542	10612	12708	2096	7468	15852	18189, 18205	11632.767
Female 50-54	76	47	12100.73684	11071	13011	1940	8161	15921	16510, 19103	11045.804
Female 55-59	37	23	12081	10333.5	13748	3414.5	5211.75	18869.75	N/A	12081
Female 60+	24	15	12586.625	11109.5	13926	2816.5	6884.75	18150.75	N/A	12586.625
Total	3217	2000								

B. Calculating Q1 and Q3 for each segment of waves

Wave Number	Swim Q1	Swim Q3	T1 Q1	T1 Q3	Cycling Q1	Cycling Q3	T2Q1	T2 Q3	Run Q1	Run Q3
1	1150.875	1330.625	228.125	254.125	3950.75	4262.375	56	72.75	2238.125	2434.5
2	1295	1592	306	482	4514	5327	111	206	3000	3831
3	1295	1592	306	482	4514	5327	111	206	3000	3831
4	1295	1592	306	482	4514	5327	111	206	3000	3831
5	1295	1592	306	482	4514	5327	111	206	3000	3831
6	1091	1307	411.5	592.5	4694.5	5443	118	211.5	2965.5	3680.5
7	1091	1307	411.5	592.5	4694.5	5443	118	211.5	2965.5	3680.5
8	1091	1307	411.5	592.5	4694.5	5443	118	211.5	2965.5	3680.5
9	1091	1307	411.5	592.5	4694.5	5443	118	211.5	2965.5	3680.5
10	1228	1566	303	482	4567	5565	101	189	2985	3751
11	1228	1566	303	482	4567	5565	101	189	2985	3751
12	1228	1566	303	482	4567	5565	101	189	2985	3751
13	1263	1561.5	299.5	464.5	4537.5	5267	108	215.5	3136	4098
14	1263	1561.5	299.5	464.5	4537.5	5267	108	215.5	3136	4098
15	1263	1561.5	299.5	464.5	4537.5	5267	108	215.5	3136	4098
16	1367	1609.5	317	510	4515	5354.5	118.5	220	3080.5	3831.5
17	1367	1609.5	317	510	4515	5354.5	118.5	220	3080.5	3831.5
18	1367	1609.5	317	510	4515	5354.5	118.5	220	3080.5	3831.5
19	1367	1609.5	317	510	4515	5354.5	118.5	220	3080.5	3831.5
20	1367	1609.5	317	510	4515	5354.5	118.5	220	3080.5	3831.5
21	1327	1616	326.5	508	4601	5324.5	135	239.5	3259	4007
22	1327	1616	326.5	508	4601	5324.5	135	239.5	3259	4007
23	1327	1616	326.5	508	4601	5324.5	135	239.5	3259	4007
24	1141.5	1324.5	439	666.5	4677	5725	133	275.5	3245	4083
25	1141.5	1324.5	439	666.5	4677	5725	133	275.5	3245	4083
26	1272.25	1525.75	397.75	645.25	5098.75	6049.5	115.75	235	3353.75	4299.75
27	1204	1431	434.5	666	5222.5	6332.5	136.5	253	3182	4016
28	1204	1431	434.5	666	5222.5	6332.5	136.5	253	3182	4016
29	1204	1431	434.5	666	5222.5	6332.5	136.5	253	3182	4016
30	1187	1425	433	669	5240	6288	130	262	3143	3941
31	1187	1425	433	669	5240	6288	130	262	3143	3941
32	1119.5	1315.5	462	718.5	4949.5	5979	135.5	283	3394.5	4630.5
33	1207	1412	451	695	5256	6328	137	266	3351	4154
34	1214.5	1471.5	461	713	5279	6520.5	140	276	3246	3996
35	1214.5	1471.5	461	713	5279	6520.5	140	276	3246	3996
36	1227.5	1445	434	673.75	5219	6685.5	126.25	263.5	3306.25	3934.75
37	1227.5	1445	434	673.75	5219	6685.5	126.25	263.5	3306.25	3934.75
38	1298.75	1540.25	389.25	610.25	4991	6228.5	155	284.75	3532.75	4707.5
39	1263.5	1477	527.75	771	5634.5	6892.5	183.75	312.5	3761	5224

C. Calculating Average Total Time by division and Linear Approximations

	# of People	# of People (2000)	Groups	Avg. Without Outliers	Slope (Average speed)	Average Linear Function
Male Pro TOTAL	7	4		6886.143	7.479	
Female Pro TOTAL	6	4		7407.8333	6.952	
Male Premier TOTAL	49	30		7842	6.567	
Female Premier TOTAL	18	11	1	8629.1667	5.968	y=5.968x
Male 30-34	362	225	4	10355.17429	4.973	y=4.973x
Male 40-44	356	221	4	10357.02865	4.972	y=4.972x
Male 25-29	246	153	3	10465.89669	4.921	y=4.921x
Male 45-49	283	176	3	10467.9058	4.920	y=4.920x
Male 35-39	385	239	5	10517.12834	4.897	y=4.897x
Male 50-54	220	137	3	10673.2654	4.825	y=4.825x
Male 55-59	155	96	2	10973.96	4.693	y=4.693x
Female 50-54	76	47		11045.804	4.662	
Male 18-24	36	22	1	11147.41667	4.620	y=4.620x
Female 30-34	209	130	3	11522.114	4.470	y=4.470x
Female 25-29	134	83	2	11538.85821	4.463	y=4.463x
Male 60+	104	65	1	11558.13592	4.456	y=4.456x
Female 45-49	118	73	1	11632.767	4.427	y=4.427x
Female 35-39	145	90	2	11706.697	4.399	y=4.399x
Female 40-44	132	82		11709.465	4.398	
Female 18-24	23	14	2	11798.6087	4.365	y=4.365x

CLY TOTAL	60	37		11805.31667	4.362	
Female 55-59	37	23	1	12081	4.263	y=4.263x
Female 60+	24	15		12586.625	4.092	
ATH TOTAL	32	20	1	13376.77419	3.850	y= 3.850x
TOTALS			39			

5.2 Codes

Code 1

```
#include <iostream>
#include <fstream>
#include <string>

using namespace std;
int a[6000][9];
int age[10][6000];
int maxx = 0;

int main() {
   ofstream fout ("RunTotalFile.out");
   ifstream fin ("TriDataSet.in");
   for (int i=1; i<=5500; i++)
   {
       for (int j=0; j<=8; j++)
            a[i][j] = 0;
     }

   for (int i=1; i<=3217; i++)
   {
          int t;
          fin>> t;
      int age;
      fin>>age;
      a[t][0] = age;

      string gender;
      fin>>gender;
      if (gender=="M") a[t][1] = 1;
      if (gender=="F") a[t][1] = 0;

      fin>>gender;

      if (gender == "CLY" || gender =="ATH")
      {
          if (gender == "CLY") a[t][2] = 4;
          if (gender == "ATH") a[t][2] = 4;
      }
      else
```

```
        {
          string category;
          fin>>category;
          if (category == "PRO") a[t][2] = 1;
          if (category == "PREMIER") a[t][2] = 2;
          if (category == "OPEN") a[t][2] = 3;
        }

        for (int j=3; j<=8; j++)
        {
          string s;
          fin>>s;
          int time = 0;
          int tt = (s[0] - '0') * 10 + s[1] - '0';
          time += 3600 * tt;
          tt = (s[3] - '0') * 10 + s[4] - '0';
          time += 60 * tt;
          tt = (s[6] - '0') * 10 + s[7] - '0';
          time += tt;
          a[t][j] = time;
        }

    }
//male Pro
fout<<"male Pro:"<<endl;

int numm = 0;
for (int i=1; i<=5500; i++)
{
            if (a[i][0]!=0 && a[i][2] == 1 && a[i][1] == 1)
            {  fout<<a[i][7]<<endl; numm++;}
}

maxx += numm;
cout<<"male Pro num:"<<numm<<endl;

//female Pro
fout<<"female Pro:"<<endl;
numm = 0;
for (int i=1; i<=5500; i++)
{
            if (a[i][0]!=0 && a[i][2] == 1 && a[i][1] == 0)
                {fout<<a[i][7]<<endl; numm++;}
}
maxx += numm;
cout<<"female Pro:"<<numm<<endl;

//male Pre
fout<<"male Pre:"<<endl;
 numm = 0;
for (int i=1; i<=5500; i++)
{
            if (a[i][0]!=0 && a[i][2] == 2 && a[i][1] == 1)
      {fout<<a[i][7]<<endl; numm++;}
```

```
}
maxx +=  numm;
cout<<"male Pre:"<<numm<<endl;

//female Pre
fout<<"female Pre:"<<endl;
numm = 0;
for (int i=1; i<=5500; i++)
{
            if (a[i][0]!=0 && a[i][2] == 2 && a[i][1] == 0)
    {fout<<a[i][7]<<endl; numm++;}
}
maxx += numm;
cout<<"female Pre:"<<numm<<endl;

//CLY
fout<<"cly:"<<endl;
numm = 0;
for (int i=1; i<=5500; i++)
{
            if (a[i][0]!=0 && a[i][2] == 4 && a[i][1] == 1)
     {fout<<a[i][7]<<endl; numm++;}
}
maxx += numm;
cout<<"cly:"<<numm<<endl;

//ATH
fout<<"ath:"<<endl;
numm = 0;
for (int i=1; i<=5500; i++)
{
            if (a[i][0]!=0 && a[i][2] == 4 && a[i][1] == 0)
    {fout<<a[i][7]<<endl; numm++;}
}
maxx+=numm;
cout<<"ath:"<<numm<<endl;


for (int i=1; i<=9; i++)
            age[i][0] = 0;

for (int i=1; i<=5500; i++)
{
            if (a[i][0]!=0 && a[i][2]==3)
            {
      int k = a[i][0] / 5 - 3;
      if (k==0) k = 1;
      if (k>9) k = 9;
      age[k][0]++;
      age[k][age[k][0]] = i;
            }
}

for (int i=1; i<=9; i++)
```

```
{
        fout<<"age group "<<i<<" female:"<<endl;
        int nn = 0;
        for (int j=1; j<=age[i][0]; j++)
        {
                int id = age[i][j];
                if (a[id][1]==0) { fout<<a[id][7]<<endl; nn++;}
        }
        cout<<"num" <<i<<": "<<nn<<endl;
  maxx +=nn;
        fout<<"age group "<<i<<" male:"<<endl;
        nn = 0;
        for (int j=1; j<=age[i][0]; j++)
        {
                int id = age[i][j];
                if (a[id][1]==1) { fout<<a[id][7]<<endl; nn++;}
        }
        maxx += nn;
        cout<<"num" <<i<<": "<<nn<<endl;

}
cout<<maxx<<endl;
  return 0;
}
```

Code 2

```
#include <iostream>
#include <fstream>
#include <string>
#include <math.h>
using namespace std;
double a[6000][9];
double age[10][6000];
int maxx = 0;

double data[39][11];
double k[39][11];
ofstream fout ("overlaps.out");
int cost;

void overlaps(int y1, int y2)
{
        cout<<y1<<","<<y2<<" overlaps:"<<endl;;
  // overlap swimming
        cout<<"swimming: ";

        if (150 + data[y2][1] < data[y1][2]) {
                cout<<"yes"<<endl;
                cost++;
```

```
        }
        else
                cout<<"no"<<endl;


        //overlap t1
        cout<<"t1: ";
    int b1 = data[y1][2];
    int e1 = data[y1][2] + data[y1][4];
    int b2 = 150 + data[y2][1];
    int e2 = 150 + data[y2][1] + data[y2][3];

    if (e1<b2 || e2<b1)
        cout<<"no"<<endl;
    else
    {
        cout<<"yes"<<endl;
        cost++;
        cout<<"the length is: ";
        int len;
        if (b1<=b2 && b2<=e1 && e1<=e2)
                len = e1 - b2;
        else if (b2<=b1 && b1<=e2 && e2<=e1)
                len = e2-b1;
        else if (b1<=b2 && b2<=e2 && e2<=e1)
                len = e2 - b2;
        else
                len = e1 - b1;

        cout<<len<<endl;
    }

        //overlap cycling
    cout<<"cycling: ";
    int y1ecyc = data[y1][2] + data[y1][4] + data[y1][6];
        int y2ecyc = 150 + data[y2][1] + data[y2][3] + data[y2][5];
        int y1bcyc = data[y1][2] + data[y1][4];
        int y2bcyc = 150 + data[y2][1] + data[y2][3];
        if (y2ecyc < y1ecyc && y2bcyc >= y1bcyc )
                {cout<<"yes"<<endl; cost++;}
        else
                cout<<"no"<<endl;

        //overlap t2
    cout<<"t2: ";
    b1 = data[y1][2] + data[y1][4] + data[y1][6];
    e1 = b1 + data[y1][8];
    b2 = 150 + data[y2][1] + data[y2][3] + data[y2][5];
    e2 = b2 + data[y1][7];

    if (e1<b2 || e2<b1)
        {cout<<"no"<<endl; cost++;}
    else
    {
```

```
            cout<<"yes"<<endl;
            cout<<"the length is: ";
            int len;
            if (b1<=b2 && b2<=e1 && e1<=e2)
                        len = e1 - b2;
            else if (b2<=b1 && b1<=e2 && e2<=e1)
                        len = e2-b1;
            else if (b1<=b2 && b2<=e2 && e2<=e1)
                        len = e2 - b2;
            else
                        len = e1 - b1;

            cout<<len<<endl;
    }

            //overlap running
    cout<<"running: ";
    int y1erun = data[y1][2] + data[y1][4] + data[y1][6] + data[y2][8] + data[y2][10];
            int y2erun = 150 + data[y2][1] + data[y2][3] + data[y2][5] + data[y2][7] + data[y2][9];
            int y1brun = data[y1][2] + data[y1][4] + data[y1][6] + data[y2][8];
            int y2brun = 150 + data[y2][1] + data[y2][3] + data[y2][5] + data[y2][7];

            if (y2erun < y1erun && y2brun >= y1brun)
                        {cout<<"yes"<<endl; cost++;}
            else
                        cout<<"no"<<endl;

            cout<<endl;

            cout<<"cost="<<cost<<endl;


}

int main() {
            ifstream fin;
            fin.open("TriDataSet.in");

    for (int i=1; i<=5500; i++)
    {
        for (int j=0; j<=8; j++)
            a[i][j] = 0;
      }

    for (int i=1; i<=3217; i++)
    {
            int t;
            fin>> t;
        int age;
        fin>>age;
        a[t][0] = age;

        string gender;
        fin>>gender;
```

```
        if (gender=="M") a[t][1] = 1;
        if (gender=="F") a[t][1] = 0;

        fin>>gender;

        if (gender == "CLY" || gender =="ATH")
        {
            if (gender == "CLY") a[t][2] = 4;
            if (gender == "ATH") a[t][2] = 4;
        }
        else
        {
          string category;
          fin>>category;
          if (category == "PRO") a[t][2] = 1;
          if (category == "PREMIER") a[t][2] = 2;
          if (category == "OPEN") a[t][2] = 3;
        }
        for (int j=3; j<=8; j++)
        {
          string s;
          fin>>s;
          int time = 0;
          int tt = (s[0] - '0') * 10 + s[1] - '0';
          time += 3600 * tt;
          tt = (s[3] - '0') * 10 + s[4] - '0';
          time += 60 * tt;
          tt = (s[6] - '0') * 10 + s[7] - '0';
          time += tt;
          a[t][j] = time;
        }

    }
            fin.close();
//          fin.clear();
            fin.open("Overlaps.in");
            for (int i=1; i<=39; i++)
                        for (int j=1; j<=10; j++)
                        {
                                    fin>>data[i][j];
                        }
            /*for (int i=1; i<=39; i++)
                        for (int j=1; j<=10; j++)
                                    cout<<data[i][j];
*/
            for (int i=1; i<=39; i++)
                        for (int j=1; j<=2; j++)
                           k[i][j] = 1500.0 / (log(data[i][j]+1)); // find k of y = kln(x)

            for (int i=1; i<=39; i++)
            for (int j=5; j<=6; j++)
                        k[i][j] = 40 * 1000 / data[i][j]; // find k of y = kx

            for (int i=1; i<=39; i++)
```

```
            for (int j=9; j<=10; j++)
                k[i][j] = 10000.0 / (exp(data[i][j]));// find k of y = ke^x

        //Assume wave 1 starts at time 0
        for (int i=1; i<=38; i++)
            overlaps(i,i+1);

  return 0; }
```

6. References (Works Cited)

@Access_Magazine. "A Bathtub Model of Downtown Traffic Congestion - ACCESS Magazine." *ACCESS Magazine*. N.p., 22 Aug. 2016. Web. 13 Nov. 2016.

Center, University Of California Transportation, and Uctc-Fr-2011-18. *September 2011A BATHTUB MODEL OF TRAFFIC CONGESTION* (n.d.): n. pag. Web.

Economics, Management and Optimization in Sports (2004): 225-33. Web.

Heazlewood, Timothy. "Prediction Versus Reality: The Use of Mathematical Models to Predict Elite Performance in Swimming and Athletics at the Olympic Games." *Journal of Sports Science & Medicine*. Asist Group, Dec. 2006. Web. 13 Nov. 2016.

"How and Why the Temperature Varies." *How and Why the Temperature Varies*. N.p., n.d. Web. 13 Nov. 2016.

Interquartile Range." *Wikipedia*. Wikimedia Foundation, n.d. Web. 13 Nov. 2016.

Liu, Yuanlong. "Track and Field Performance Data and Prediction Models: Promises and Fallacies."

"Motion Graphs." *Motion Graphs*. N.p., n.d. Web. 13 Nov. 2016.

Prendergast, Kevin. *A MATHEMATICAL MODEL OF THE 100M AND WHAT IT MEANS* (n.d.): n. pag. Web.

"Schedule - Lake Tahoe Triathlon." *Lake Tahoe Triathlon*. N.p., n.d. Web. 13 Nov. 2016.

"Sprinters vs Distance Swimmers." *Feel For The Water! Advice & Tips to Improve Your Swimming.:*. N.p., 01 Jan. 1970. Web. 13 Nov. 2016.

"Statistics and Probability Dictionary." *Interquartile+range: Definition*. N.p., n.d. Web. 13 Nov. 2016.

"Triathlon." *Wikipedia*. Wikimedia Foundation, n.d. Web. 13 Nov. 2016.

"Traffic Congestion." *Wikipedia*. Wikimedia Foundation, n.d. Web. 13 Nov. 2016.

"Triathlon Swim Starts: Wave vs. Time Trail (and Others)." *Complete Tri*. N.p., 03 Sept. 2016. Web. 13 Nov. 2016.