Problem 1) (15 points) I am working for a Russian company and have been given the following problem. The given tasks must be done by their employees on a BOSS coding problem: code anchors, code boats, code storms, code zebras. A table of expected times that it would take each employee to do a particular job is given below, along with wages paid:

<table>
<thead>
<tr>
<th></th>
<th>Code Anchors</th>
<th>Code Boats</th>
<th>Code Storms</th>
<th>Code Zebras</th>
<th>Wages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tominsky</td>
<td>6</td>
<td>4</td>
<td>3</td>
<td>10</td>
<td>$40/hr</td>
</tr>
<tr>
<td>Billinsky</td>
<td>8</td>
<td>6</td>
<td>5</td>
<td>12</td>
<td>$50/hr</td>
</tr>
<tr>
<td>Jackinsky</td>
<td>9</td>
<td>8</td>
<td>Can’t code storms</td>
<td>Can’t code zebras</td>
<td>$20/hr</td>
</tr>
<tr>
<td>Sueinsky</td>
<td>11</td>
<td>10</td>
<td>9</td>
<td>11</td>
<td>$30/hr</td>
</tr>
</tbody>
</table>

Assuming that everyone must work and that their code is independent of each other, they have asked me to set up the Russian LP solution to solve for

a) The most expensive way to assign the jobs,
b) The longest time to finish the job.

Yea, yea. Don’t tell me your problems. I just work here. I guess that’s just the way they do business. Do it.

Problem 2) (15 points) An annuity works like this: I put in $1000/month for 30 years, and then retire. At the end of the next month I then start withdrawing a monthly amount of money from the bank, an annuity, to live on. If I expect to live 10 years after retirement, how much will I have available to live on each month if i = 6% per year?

Problem 3) (30 points) Write a boss simulation to study the following system: I have a reservoir with a capacity of 2 million acre feet of water. Water enters the reservoir from 3 sources as follows:

<table>
<thead>
<tr>
<th>Source</th>
<th>Distribution</th>
<th>Min volume/day</th>
<th>Max volume/day</th>
<th>Starts flowing</th>
<th>Quits flowing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yegua Creek</td>
<td>Normal</td>
<td>2000 acre ft/day</td>
<td>4000 acre ft/day</td>
<td>January 1</td>
<td>April 30</td>
</tr>
<tr>
<td>Yegua Creek</td>
<td>Normal</td>
<td>3000 acre ft/day</td>
<td>6000 acre ft/day</td>
<td>May 1</td>
<td>December 30</td>
</tr>
<tr>
<td>Birch Creek</td>
<td>Uniform</td>
<td>1000 acre ft/day</td>
<td>3000 acre ft/day</td>
<td>July 1</td>
<td>October 30</td>
</tr>
<tr>
<td>Rocky Creek</td>
<td>Exponential</td>
<td>3000 acre ft/day</td>
<td>3000 acre ft/day</td>
<td>All year round</td>
<td>All year round</td>
</tr>
</tbody>
</table>

Water is removed from the reservoir through five 60” diameter pipes and pumped to Houston at a constant rate of 4000 acre ft/day, when water is available. If the reservoir drops below 100,000 acre feet we stop pumping and send nothing that day.

I am currently designing the spillway and need to know the maximum flow rate/day over the spillway. You may assume that there are 30 days in each month and 360 days in the year. Please start your simulation with the reservoir half full, and study it for 5 years. I also need to know the number of days during that 5 years that we send no water to Houston.
Problem 4) (20 points) For the program listed below, determine the most likely final values of counter1 and counter2:

```plaintext
PROGRAM
  "Check code"
DEFINITION
  ATTRIBUTES = {counter1 = 10};
  counter2 = 20;
CONTROL
  STOPTIME=100;
  RANDOMIZE = ON;
LOGIC
  ARRIVE {TIME=EXPD(5)};
    counter1 = counter1 + 1;
    counter2 = counter2 + 1;
  DEPART{};
END.
```

Problem 5) (20 points) For the program listed below, determine the most likely final values of turkey and giblets:

```plaintext
PROGRAM
  "Check code"
DEFINITION
  giblets = 10;
  turkey = 10;
  LABELS={pig,dog,cat};
CONTROL
  STOPTIME=1000;
  RANDOMIZE = ON;
LOGIC
  cat: giblets = giblets -5;
    IF turkey >= 20 then GOTO pig ELSE GOTO dog;
    ARRIVE {TIME=EXPD(5), LIMIT = 2};
  dog: giblets = giblets - 1;
    turkey = turkey + 3;
    IF giblets >= 9 THEN GOTO cat ELSE giblets = 0;
  pig: DEPART{};
END.
```
Problem 1) (20 points) Find and list on a separate sheet any errors in the code below. DO NOT MAKE ANY MARKS ON THIS PAGE. On the following sheet, simply list the line number and the error involved. You do not have to copy down the line of code. Simply note its line number. The line numbers are not listed sequentially.

1 PROGRAM
2 "computer lab"
3 DEFINITION
4 leftmad=0;
5 runtime = 6.0;
6 mytime = 0.0;
7 engineercost = 50/60
8 ATTRIBUTES = {mytime=0.0};
9 pc: RESOURCE = {CAPACITY=15};
10 LABELS = {cat, goawaymad, wait};
11 CONTROL
12 STOPTIME=8*60;
13 RANDOMIZE=ON;
14 LOGIC
15 ARIIVE {TIME = EXPD(12)};
16 mytime=CLOCKTIME;
17 SIEZE{NAME=pc};
18 WAIT{TIME = 0 MAX NORMAL(2*runtime, 0.1*2*runtime)};
19 RELEASE{NAME=pc};
20 cat: IF totalcost <= 5000 THEN GOTO cat ELSE GOTO goawaymad;
21 totalcost = totalcost + engineercost;
22 DEPART{ }; 
23 ARRIVE {TIME = EXPD(1.2)};
24 mytime=CLOCKTIME;
25 SIEZE{NAME=pc};
26 IF CLOCKTIME >= 60 GOTO dog ELSO GOTO wait;
27 WAIT{TIME = 0 MAX NORMAL(runtime, 0.1*runtime)};
28 RELEASE{NAME=pc};
29 totalcost = totalcost + 3*engineercost;
30 DEPART{ }; 
31 goawaymad: leftmad = leftmad + 1;
32 totalcost = totalcost + 10*engineercost;
33 IF totalcost >= 20000 THEN GOTO wait ELSE GOTO cat;
34 DEPART{ }; 
35 dog: ARRIVE{TIME = EXPD(60)};
36 mytime=CLOCKTIME;
37 SIEZE{NAME=pc};
38 WAIT TIME = 0 MAX NORMAL(4*runtime, 0.1*4*runtime);
39 wait: RELEASE{NAME=pc};
40 totalcost = totalcost + 2 x engineercost;
41 DEPART{ }; 
42 END.
<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description of error</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>No, I don’t know how many there are. If you need more room just write them below here.</td>
</tr>
</tbody>
</table>
Problem 2) (25 points) I have decided to give my 422 final exam in Room 140. There are only 50 seats in that room, so I intend to spread the exam out over the day, starting at 8 am, allowing the students to come at a time of their choice, and leave when they finish the exam. Fifty students have agreed to come sometime during the morning, from 8 to 12 and they are expected to arrive exponentially distributed with a mean of 3 minutes. The remaining students have agreed to come between 12 pm and 3 pm, and will arrive normally distributed with a mean of 4 minutes, and a sigma of 1 minute. The exam should only take them 40 minutes to finish, normally distributed with a sigma of 10 minutes.

Write a BOSS simulation from which I can determine the approximate maximum number of students who will have to stand in line waiting for a seat during this exam. Also, include code to tell me the approximate time that the last student will finish and I can go home.

Problem 3) (30 points) Write a boss simulation to study the following system: My company makes custom crankshafts for motorcycles, including Orange County Choppers. The shafts come out of a casting machine pretty much ready to use, although some come out too large and some too small. They come out of the casting process with a nominal diameter of 2.0000”, normally distributed with a sigma = 0.0010”

Just Right Cranks (within tolerances): These have a diameter from 1.9980” to 2.0010” These are wrapped, put in a box and shipped to the customer. A basic crank costs us $800 to make, if no secondary operations are required, i.e. if it is within tolerances straight out of the casting machine.

Too Big Cranks: Cranks above 2.0010” have to undergo a secondary process of grinding, wherein the diameter is ground down (nominally) 0.0002” per pass, until the shaft is within tolerance. This grinding operation reduces the diameter of the shaft by 0.0002” with a sigma = 0.0001” per pass. Sometimes it is possible that the grinding operation may actually make the shaft too small, at which time the crank must go in with the “Too Small Cranks”. This grinding operation costs us $20/pass, so if it takes you 3 grindings to get the shaft down to within specs, expect to pay $60.

Too Small Cranks: Cranks below 1.9980” must be sent to our weld overlay shop, wherein a thin weld is built up around the diameter of the crank. This weld is variable, averaging 0.0030” thick, with a sigma of 0.0010”. This operation costs us $60/pass and continues until the shaft diameter is above 2.0010”, at which time it is sent for grinding with the other Too Big Cranks.

Write a BOSS program to determine the approximate total cost for us to manufacture 100 crankshafts under these conditions.

Problem 4) (25 points) For the following cash flow diagram, determine A if i = 8%, and F = $30,000.