Delayed Discounting Computer Program 10/7/08

I) As the experiment file is set up, there are several things that we want to be standard in all records. They are as follows:

   i. Input:
      1. Experiment number (we input in the experiment file)
      2. Student ID number (participant enters)
      3. Gender (participant checks box for "male" or "female")
      4. Age (participant enters)
      5. Instructions (text that we input in experiment file)

   ii. Output:
      1. Experiment number
      2. Student ID number (participant enters)
      3. Gender – coded as "1" for male and "2" for female
      4. Age
      5. Start date/time (computer automatically codes)

II) Since this is similar to HTML, I assume you have a choice between radio buttons, boxes, etc. for input. I would prefer boxes because radio buttons have a default choice whereas boxes do not, and I do not want there to be a preset choice. However, they should only be able to check one and not several. Ideally, the program should record choices numerically, so I don't have to recode words. Thus, it should record the values of the points chosen only (not the additional text presented on the screen).

III) In this computer program, we will want several "choice options" which represent different modules that can be added or deleted to the experiment file, based on the experiment that we want to run. As we had previously talked about, there are several different variations of this study that we want to run and we want the program to have maximum flexibility. There are five different "choice options" that we want to be able to include.

Choice Option 1. The <u>double limit method</u>. This choice option assesses discounting using the double-limit method algorithm to converge on a consistent indifference point.

   i. The prompt on each screen should read: "Would you prefer" in the middle to upper left. A bit below that should be two choices – one on the left and one on the right.

   ii. The adjusting outcome needs to be presented on the left with the words "*x* points right now" (with *x* representing the amount produced by the programmed algorithm.

   iii. The standard outcome (the maximum points/time) should be presented on the right with the words with the words "*y* points in *z* minutes" (*y* representing the maximum points and *z* representing delay time in order to receive maximum points).

   iv. After the two choices are presented, there should be a possible delay (based on the input in #4 below) before the boxes appear that allow the person to make his/her choice. This is to make sure they read the choices and don't just rush through.

v. Below the boxes, there needs to be a "submit choice" button such that the data are not recorded until that button is pressed (and clicking on that button stops the reaction time counter)

vi. Reaction time should be calculated starting with the presentation of the choices (when the screen first appears) until they push the submit button.

vii. Input would consist of:

   1. Minimum points
   2. Maximum points
   3. Delay time in minutes to receive maximum points
   4. Time in seconds between the presentation of the choices and the presentation of the boxes for the person to click on to make their choice

viii. Algorithm: The first choice the participant makes is between the delayed amount and a probabilistically selected adjusting amount that is between 0% and 100% (using multiples of 2%) of the value of the delayed amount. Following each choice, an adjusting outcome is probabilistically selected from a range of values that is narrowed when consistent choices are made by the participant. This algorithm continues until the participant responds consistently within a narrow range of values that correspond to a single indifference point. The indifferent point is calculated by taking the value of adjusting outcome amount after the algorithm has converged. Details of the algorithm are located on pp. 144-146 of Johnson & Bickel (2002).

ix. Output would consist of:

   1. Minimum points
   2. Maximum points
   3. Delay time in minutes to receive maximum points
   4. Number of trials to reach indifference point
   5. Choice option identified as "1"
   6. Time in seconds between the presentation of the choices and the presentation of the boxes for the person to click on to make their choice
   7. Choice made by participant on each trial
   8. Reaction time in seconds for each trial
   9. Total time elapsed for task (full algorithm to be run) in seconds

Choice Option 2. The <u>decreasing adjustment method.</u> This choice option uses the decreasing adjustment method to converge on a single indifference point in a set number of trials.

i. The prompt on each screen should read: "Would you prefer" in the middle to upper left. A bit below that should be two choices – one on the left and one on the right.

ii. The adjusting outcome needs to be presented on the left with the words "*x* points right now" (with *x* representing the amount produced by the programmed algorithm.

iii. The standard outcome (the maximum points/time) should be presented on the right with the words with the words "*y* points in *z* minutes" (*y* representing the maximum points and *z* representing delay time in order to receive maximum points).

iv. After the two choices are presented, there should be a possible delay (based on the input in #4 below) before the boxes appear that allow the person to make his/her choice. This is to make sure they read the choices and don't just rush through.

v. Below the boxes, there needs to be a "submit choice" button such that the data are not recorded until that button is pressed (and clicking on that button stops the reaction time counter)

vi. Reaction time should be calculated starting with the presentation of the choices (when the screen first appears) until they push the submit button.

vii. Input would consist of:

   1. Minimum points
   2. Maximum points
   3. Delay time in minutes to receive maximum points
   4. Time in seconds between the presentation of the choices and the presentation of the boxes for the person to click on to make their choice
   5. Number of trials to reach indifference point (we set this in this method)

viii. Algorithm: This algorithm would consist of having the participant first choose between the delayed amount and exactly half of the delayed amount. Following each choice the value of the adjusting outcome is increased or decreased by half of the previous adjustment, depending on the choice of the participant. The procedure would continue until a specified number of trials have elapsed (e.g., 6), at which time the value that would have been presented as the adjusting outcome on the next trial is accepted as the indifference point.

ix. Output would consist of:

   1. Minimum points
   2. Maximum points
   3. Delay time in minutes to receive maximum points
   4. Time in seconds between the presentation of the choices and the presentation of the boxes for the person to click on to make their choice
   5. Number of trials to reach indifference point
   6. Choice option identified as "2"
   7. Choice made by participant on each trial
   8. Reaction time in seconds for each trial
   9. Total time elapsed for task (full algorithm to be run) in seconds

Choice Option 3. <u>Multiple-choice (ruler) assessment.</u> Participants are given a set number of possible choices and they choose a single indifference point in one trial. This option would be the one that looks like the rulers that I showed you. We could run it two possible ways – one only varying the points (thus, it would work like the options laid out above, but they would only make a single choice as to the fewest number of points they would take now rather than wait for the full points, or it could be run like the last study with the rulers – where students could earn a proportion of points for a proportion of the amount of time.

i. The prompt on each screen should read: "Would you prefer" in the middle to upper left. A bit below that should be the list of choices. In this option, the best placement of choices would be in list format with the standard choice (maximum points/time) at the top and the smaller-sooner choices listed in descending order with boxes on the left side for them to check their choice.

ii. In this case, they would see a list that says "*y* points in *z* minutes" (*y* representing the points and *z* representing delay time in order to receive points for each ordered pair – starting with the maximum values working down to the minimum values).

iii. After the choices are presented, there should be a possible delay (based on the input in #4 below) before the boxes appear that allow the person to make his/her choice. This is to make sure they read the choices and don't just rush through.

iv. Below the boxes, there needs to be a "submit choice" button such that the data are not recorded until that button is pressed (and clicking on that button stops the reaction time counter)

v. Reaction time should be calculated starting with the presentation of the choices (when the screen first appears) until they push the submit button.

vi. Input would consist of:
   1. Minimum points (to be recorded directly in output file)
   2. Maximum points (to be recorded directly in output file)
   3. Delay time in minutes to receive maximum points (to be recorded directly in output file)
   4. Time in seconds between the presentation of the choices and the presentation of the boxes for the person to click on to make their choice
   5. Listing of each possible point/time option as ordered pairs
   6. Points distribution (e.g., linear, exponential, log) input as a data table

vii. Algorithm: The program would record which choice each participant made – no calculations are necessary.

viii. Output would consist of:
   1. Minimum points
   2. Maximum points
   3. Delay time in minutes to receive maximum points (to be recorded directly in output file)
   4. Time in seconds between the presentation of the choices and the presentation of the boxes for the person to click on to make their choice
   5. Choice option identified as "3"

6. Choice made by participant on each trial recorded as an ordered pair
7. Points distribution used (e.g., "1" linear, "2" exponential, "3" log)
8. Reaction time in seconds for each trial
9. Total time elapsed for task in seconds

We're still working the kinks out on #4 and #5, but I thought this would give you enough to start with.

Choice Option 4. Real-time assessment. An indifference point is determined based on where the student chooses to stop in a set-length task.

Choice Option 5. Escape condition. Participants begin a task whose length is set based on a previous choice the student has made (most likely in completing one of the choice options above). They may either complete the task or escape from the task at any point. Elapsed time before escaping the task would serve as the indifference point.

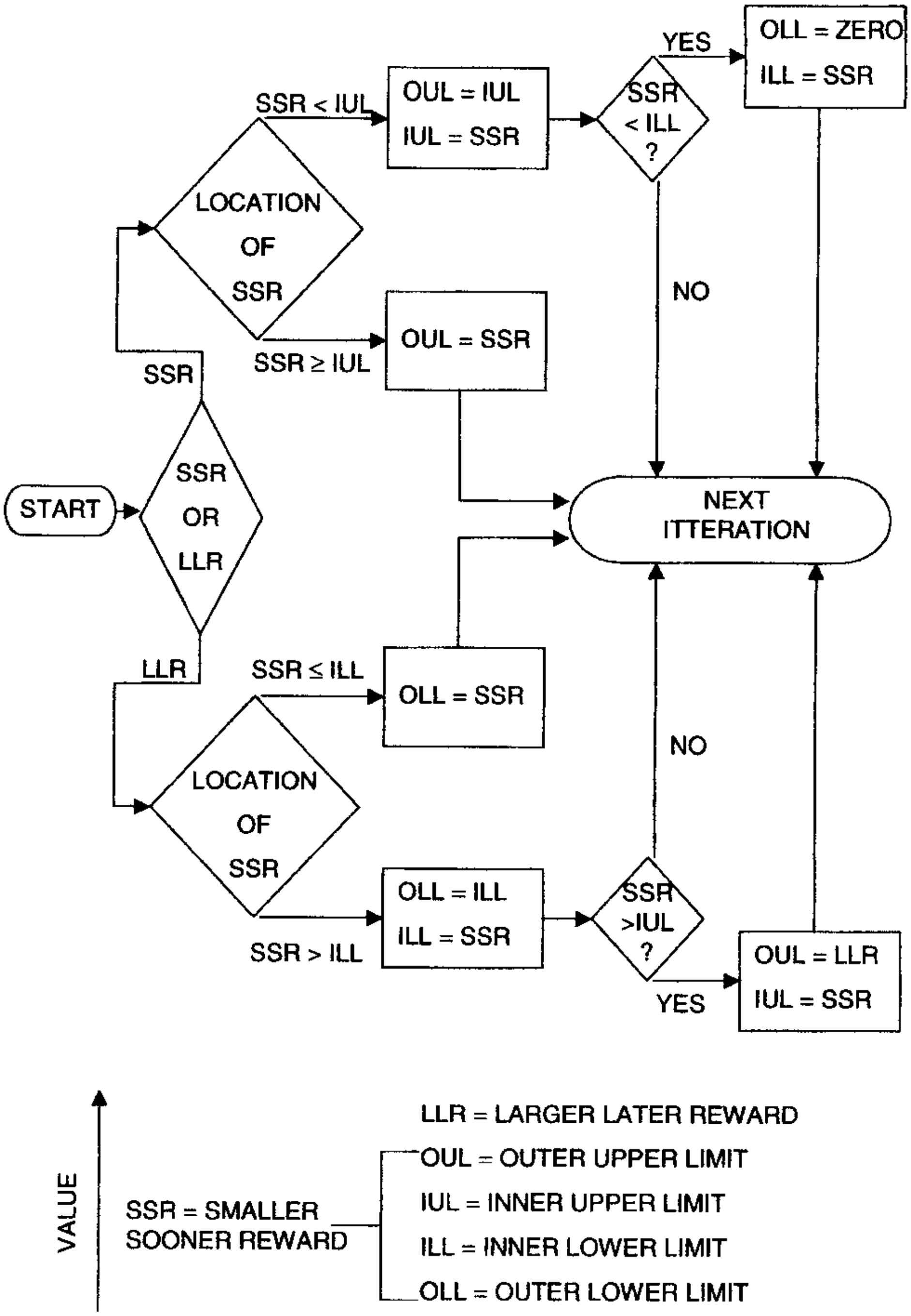

Fig. 9. Choice algorithm used to locate indifference points. Rectangles represent a process in which the variable on the left side of the equation takes the value of the variable on the right side. Diamonds indicate differential outcomes depending on the condition inside the diamond. The procedure is repeated until the difference in value between the OUL and the OLL is 2% or less of the value of the larger delayed reward.

```xml
<?xml version="1.0"?>

<experiment xmlns="http://www.w3schools.com"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xsi:schemaLocation="experiment.xsd">

    <personnel>
        <investigator>
            <name> Heidi Dempsey </name>
            <department> Psychology </department>
        </investigator>
        <investigator>
            <name> David Dempsey </name>
            <department> Mathematics </department>
        </investigator>
        <investigator>
            <name> Aaron Garrett </name>
            <department> Computer Science </department>
        </investigator>
    </personnel>

    <title> Discounting in Study Habits Experiment </title>
    <number> 1 </number>
    <instructions>
        All the instructions should go here and will be placed in the interface.
    </instructions>

    <methods>
        <!-- There are 3 different possibilities (right now) for the experiment methods: dl (double limit), da (decreasing adjustment), and mc (multiple choice). -->
        <!-- Each of these can be placed as many times and in any order that you choose. -->
        <!-- The time variables (maxTime and responseDelay) have an optional attribute called "units" that can be set to any of the following values: -->
        <!-- "seconds", "minutes", "hours", "days", or "weeks" -->
        <!-- The default value is "seconds" if it is left unspecified. -->
        <dl>
            <valueType> cookie </valueType>
            <minValue> 0.5 </minValue>
            <maxValue> 10 </maxValue>
            <maxTime units="minutes"> 2 </maxTime>
            <responseDelay units="seconds"> 3 </responseDelay>
        </dl>
        <da>
            <valueType> point </valueType>
            <minValue> 0.5 </minValue>
            <maxValue> 10 </maxValue>
            <maxTime units="minutes"> 2 </maxTime>
            <responseDelay> 1 </responseDelay>
            <numTrials> 7 </numTrials>
        </da>
        <dl>
            <valueType> vacation </valueType>
            <minValue> 0.5 </minValue>
            <maxValue> 10 </maxValue>
            <maxTime units="minutes"> 2 </maxTime>
            <responseDelay units="seconds"> 0 </responseDelay>
        </dl>
        <mc>
            <valueType> dollar </valueType>
            <minValue> 0.5 </minValue>
            <maxValue> 10 </maxValue>
            <maxTime units="minutes"> 2 </maxTime>
            <responseDelay units="seconds"> 1 </responseDelay>
            <distribution> Exponential </distribution>
            <choices>
                <item>
                    <value> 0.01 </value>
                    <time> 0 </time>
                </item>
                <item>
                    <value> 0.5 </value>
                    <time units="seconds"> 30 </time>
                </item>
                <item>
                    <value> 1 </value>
                    <time> 60 </time>
                </item>
            </choices>
        </mc>
        <mc>
            <valueType> ice cream </valueType>
            <minValue> 1 </minValue>
            <maxValue> 15 </maxValue>
            <maxTime units="minutes"> 2 </maxTime>
            <responseDelay units="seconds"> 2 </responseDelay>
            <distribution> Logarithmic </distribution>
            <choices>
                <item>
                    <value> 0.1 </value>
                    <time> 0 </time>
                </item>
                <item>
                    <value> 0.25 </value>
                    <time units="seconds"> 10 </time>
                </item>
                <item>
                    <value> 1 </value>
                    <time> 30 </time>
                </item>
            </choices>
        </mc>
    </methods>

</experiment>
```