



# **Loan Eligibility Determinator**

Presented by Christian Lussier, Trent  
Faulkner, Robert Samuel, and Mikey Spurr

December 2018

**Allegheny College**  
**CS102**

# Overview

## What actually is our project?

- Program determines if an applicant is eligible for loan based off answers to questions.
- 2 sets of questions.
  - Set of 3 Questions -- fast-track/preliminary.
    - If ineligible, person using program can move on to next set.
  - Set of 7 Questions -- in depth.
- Based off inputted answers, truth tables and boolean logic/equation used to determine eligibility.
- Option to save eligibility status to file.



# Motivation

Why did we pick this topic?

- ★ Collectively interested in looking at how we could tie in 102 into economics.
- ★ One thing that came to our mind was how banks give out loans.
  - We enjoyed the lab and classes about boolean logic & truth tables.
  - How we could implement boolean truth tables for this concept?

## ★ Real-World Importance

- Make sure people who are trying to acquire loans go through fair but reasonable questions.
- There have been many studies on the implications of bad loans.

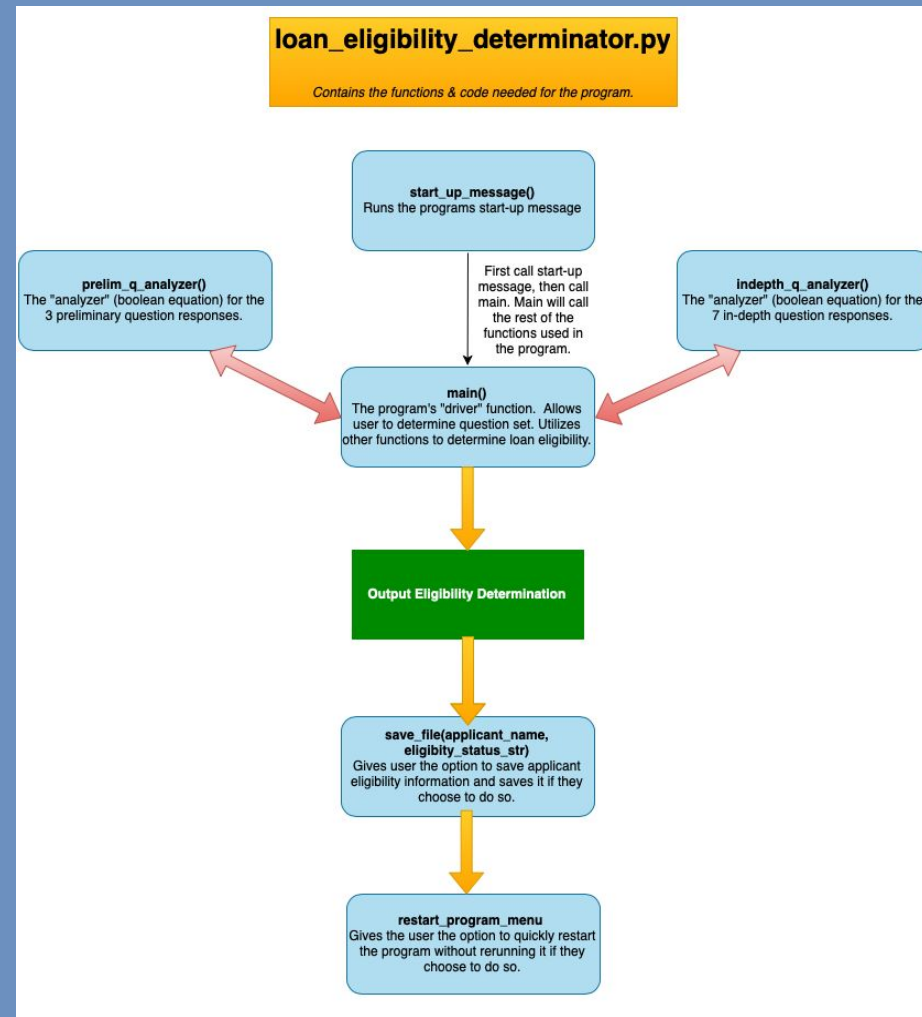
# Challenges

What challenges did we face?

- ❑ Our biggest challenge was figuring out what questions to ask.
  - ❑ For example, our questions couldn't be too broad but also couldn't be applicable to one group of people.
  - ❑ Overcame this challenge by doing research and analyzing our potential questions..
- ❑ Coming up with an accurate equation for the boolean truth table.
  - ❑ Applicants can't have two no's on major questions and still be allowed a loan because they are above the threshold.
- ❑ Lastly we came across issues trying to send the outputs into a text file.
  - ❑ At first, you had to create a new text file every time; cumbersome and annoying.

# Implementation

- Display startup messages ("print"); applicant enters name ("input").
- User enters whether or not they want to go the fast track with 3 fast-track/preliminary questions or 7 in depth questions.
- Store responses in truth table list; use boolean equation to determine eligibility.
- Finally, regardless if you get the loan or not, you have the option for your name and eligibility status to be stored into a text file.
- Can choose to restart the program for convenience.



```
Activities Terminal Mon 3:47 PM
spurr@aldenv178: ~/A_Junior_Year/cs102f2018/cs102f2018-spurr/labs/lab5/cs102f2018-lab05-starter-absolute-units/src
File Edit View Search Terminal Help
| * About The Program: This program is for checking a applicant's eligibility |
| for a loan from a bank. The banker should ask the user questions and record |
| their Yes or No answers in the program. There are two sets of questions: one |
| quick set and one longer set for more interesting cases. The program will |
| give a determination on whether the user should receive a loan. |
|-----|
What is the applicant's name? -- Mikey

Do you want to begin with 3 preliminary questions? Type YES if you do, otherwise, hit enter to skip the preliminary question set. --
Please record applicant responses to the asked questions!
*** 7 In-Depth Questions ***
1) Is the applicant in good standing with the bank? YES or NO? -- yes
2) Is the applicant's credit score above a 580 ('Fair')? YES or NO? -- yes
3) Does the applicant have enough (or close to enough) collateral for their requested loan? YES or NO? -- no
4) Is the term for loan repayment requested by the applicant realistic? YES or NO? -- yes
5) Will the applicant be using the loan for a practical/realistic purpose that won't interfere with their education? YES or NO? -- yes
6) Does the applicant currently have any unpaid debts/outstanding payments? YES or NO? -- yes
7) Is the applicant currently employed? YES or NO? -- yes
Mikey has been determined to be ELIGIBLE for the loan based off the preliminary question set.

Do you want to save the applicants's loan eligibility status to a text file? YES or NO -- yes
Do you want to restart the program? YES or NO? --- no

Closing the program!
```

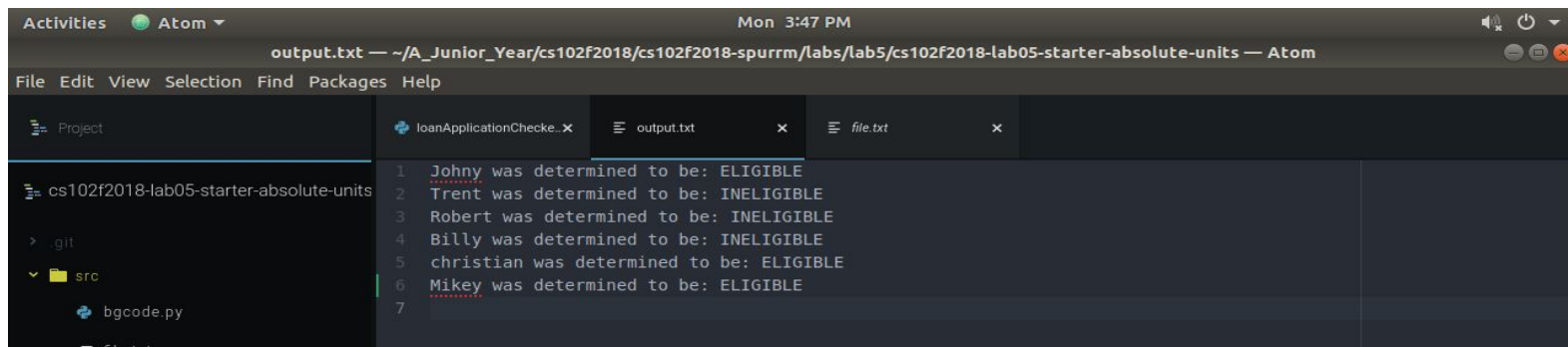
```
loan_eligibility_determinator.py
49 ~ def prelim_q_analyzer(in1_bool, in2_bool, in3_bool):
50     """The analyzer code/equation for the preliminary question set."""
51     return in1_bool and in2_bool and in3_bool
52
53
54 ~ def indepth_q_analyzer(in1_bool, in2_bool, in3_bool, in4_bool, in5_bool, in6_bool, in7_bool):
55     """The analyzer code/equation for the in-depth question set."""
56     return in1_bool and in2_bool or in3_bool and in4_bool or in5_bool and in6_bool and in7_bool
57
58
59 ~ def main():
60     """The main 'driver' function of the program."""
61     applicant_name = input("What is the applicant's name? -- ")
62     print() # spacing line for better readability.
63     which_qs = input("Do you want to begin with 3 preliminary questions? Type YES if you do, otherwise, hit enter to skip the preliminary question set. -- ")
64     print("Please record applicant responses to the asked questions!")
65
66 ~ if which_qs == "YES" or which_qs == "yes" or which_qs == "Yes": # if statement that checks if user wants to begin with 3 preliminary questions
67     print("** 3 Preliminary Questions **")
68     ## Question 1
69     q1_answer = input("1) Does the applicant present themselves well? YES or NO? -- ")
70     q1_list = [] # creates list where question 1 answer boolean will be stored.
71 ~ if q1_answer == "YES" or q1_answer == "yes" or q1_answer == "Yes":
72     q1_list = [True]
73 ~ else:
74     q1_list = [False]
75
76     ## Question 2
77     q2_answer = input("2) Does the applicant have at least a credit score of at least 700? YES or NO? -- ")
78     q2_list = [] # creates list where question 2 answer boolean will be stored.
79 ~ if q2_answer == "YES" or q2_answer == "yes" or q2_answer == "Yes":
80     q2_list = [True]
81 ~ else:
82     q2_list = [False]
83
84     ## Question 3
85     q3_answer = input("3) Does the applicant have enough (or close to enough) collateral for their requested loan? YES or NO? -- ")
86     q3_list = [] # creates list where question 3 answer boolean will be stored.
87 ~ if q3_answer == "YES" or q3_answer == "yes" or q3_answer == "Yes":
88     q3_list = [True]
89 ~ else:
90     q3_list = [False]
91
92     ## Question 4
93     q4_answer = input("4) Is the term for loan repayment requested by the applicant realistic? YES or NO? -- ")
94     q4_list = [] # creates list where question 4 answer boolean will be stored.
95 ~ if q4_answer == "YES" or q4_answer == "yes" or q4_answer == "Yes":
96     q4_list = [True]
97 ~ else:
98     q4_list = [False]
99
100     ## Question 5
101     q5_answer = input("5) Will the applicant be using the loan for a practical/realistic purpose that won't interfere with their education? YES or NO? -- ")
102     q5_list = [] # creates list where question 5 answer boolean will be stored.
103 ~ if q5_answer == "YES" or q5_answer == "yes" or q5_answer == "Yes":
104     q5_list = [True]
105 ~ else:
106     q5_list = [False]
107
108     ## Question 6
109     q6_answer = input("6) Does the applicant currently have any unpaid debts/outstanding payments? YES or NO? -- ")
110     q6_list = [] # creates list where question 6 answer boolean will be stored.
111 ~ if q6_answer == "YES" or q6_answer == "yes" or q6_answer == "Yes":
112     q6_list = [True]
113 ~ else:
114     q6_list = [False]
115
116     ## Question 7
117     q7_answer = input("7) Is the applicant currently employed? YES or NO? -- ")
118     q7_list = [] # creates list where question 7 answer boolean will be stored.
119 ~ if q7_answer == "YES" or q7_answer == "yes" or q7_answer == "Yes":
120     q7_list = [True]
121 ~ else:
122     q7_list = [False]
123
124     # Determine if applicant is eligible for loan
125     if q1_list[0] and q2_list[0] and q3_list[0] and q4_list[0] and q5_list[0] and q6_list[0] and q7_list[0]:
126         print("Mikey has been determined to be ELIGIBLE for the loan based off the preliminary question set.")
127     else:
128         print("Mikey has been determined to be NOT ELIGIBLE for the loan based off the preliminary question set.")
129
130     # Ask if user wants to save status to text file
131     save_status = input("Do you want to save the applicants's loan eligibility status to a text file? YES or NO -- ")
132     if save_status == "YES" or save_status == "yes" or save_status == "Yes":
133         # Save status to text file
134         with open("applicant_status.txt", "w") as f:
135             f.write(f"Applicant Name: {applicant_name}\n")
136             f.write(f"Eligible: {q1_list[0] and q2_list[0] and q3_list[0] and q4_list[0] and q5_list[0] and q6_list[0] and q7_list[0]}\n")
137             f.write(f"Status: {'ELIGIBLE' if q1_list[0] and q2_list[0] and q3_list[0] and q4_list[0] and q5_list[0] and q6_list[0] and q7_list[0] else 'NOT ELIGIBLE'}\n")
138         print("Status saved to applicant_status.txt")
139     else:
140         print("Status not saved to text file.")
141
142     # Ask if user wants to restart program
143     restart = input("Do you want to restart the program? YES or NO? --- ")
144     if restart == "YES" or restart == "yes" or restart == "Yes":
145         main()
146     else:
147         print("Closing the program!")
148         exit()
```



# Results & Analysis

How did our program perform?

- Program runs smooth allowing the input or lowercase and uppercase for answers, quick restart of program option, etc.
- Program does a good job of weighting question importance, but could be more fine-tuned so that it makes a more accurate determination.
  - Improve upon the boolean/truth-table determination equation!



The screenshot shows the Atom code editor interface. The top status bar indicates the current file is `output.txt` located at `~/A_Junior_Year/cs102f2018/cs102f2018-spurrmlabs/lab5/cs102f2018-lab05-starter-absolute-units`. The left sidebar displays a file tree with the project structure, including a `src` directory containing `bgcode.py`. The main editor area shows the contents of `output.txt`, which contains the following text:

```
1 Johnny was determined to be: ELIGIBLE
2 Trent was determined to be: INELIGIBLE
3 Robert was determined to be: INELIGIBLE
4 Billy was determined to be: INELIGIBLE
5 christian was determined to be: ELIGIBLE
6 Mikey was determined to be: ELIGIBLE
7
```



# *DEMONSTRATION*

In the Linux terminal...





*Any Questions?*

Thank you!