Mobile Devices

FRANKFURT UNIVERSITY OF APPLIED SCIENCES

Requirements on “Project 1”

Laboratory – Goals

- Getting familiar with Android Studio
- Developing an app with manageable requirements which uses
  - different user interface elements
    - basic methods
  - two screens / activities which require data transfer
- Logging and debugging

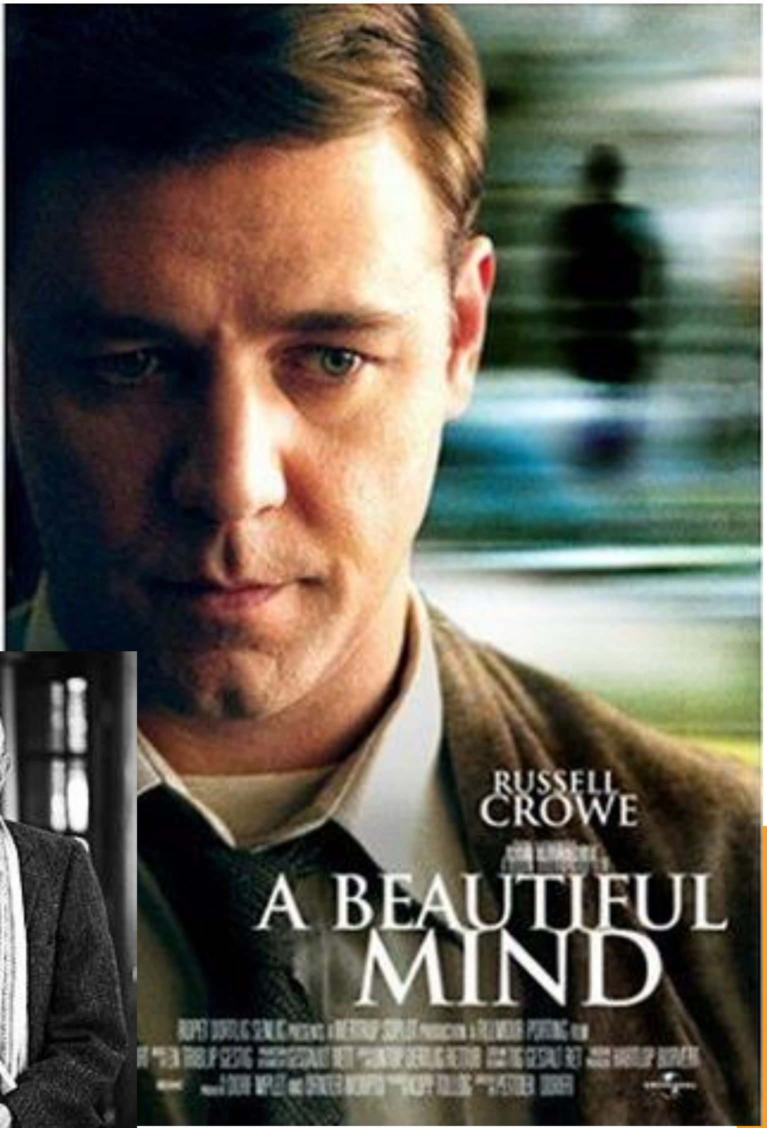
➔ To be done at home & during lab meetings!

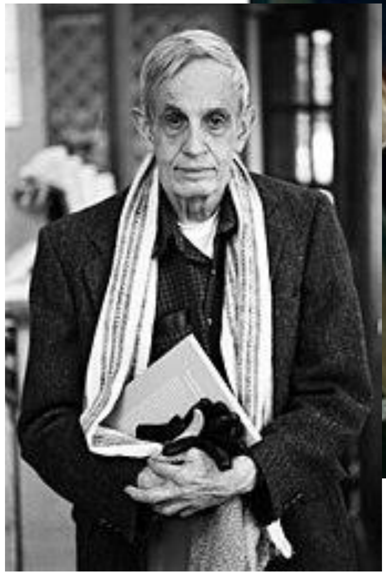
Laboratory – App “Project 1”

- Create an app called “A Beautiful Mind” according to the requirements given in this document.
  - according to your own time management
  - This is your “Project 1” of lecture Mobile Devices!
  - Use the lab meetings for bringing up your questions, for sharing ideas … etc.
- The requirements for the Project 1 are given here.

Project 1: „A Beautiful Mind“

- “A Beautiful Mind” is a 2001 American biographical drama film directed by Ron Howard and starring Russell Crowe as John Nash, a brilliant but troubled mathematician who develops schizophrenia.
- The film is based on the real-life story of Nash, who won the Nobel Memorial Prize in Economic Sciences in 1994 for his work on game theory.

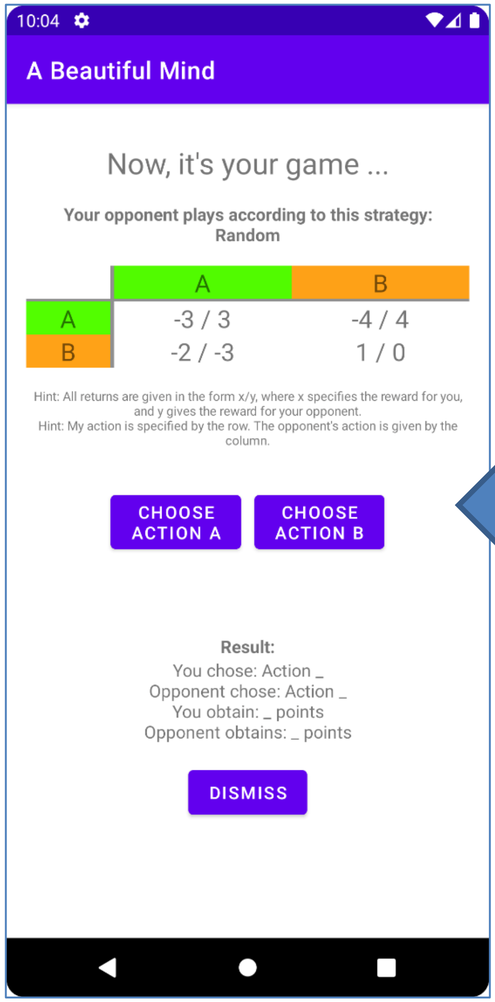




imdb.com, wikipedia.org

Project 1: „A Beautiful Mind“

- The app you develop will be an homage to that movie, to John Nash, and to game theory.
- Before we specify the requirements in detail, we must get some basic knowledge about game theory.



10:04

A Beautiful Mind

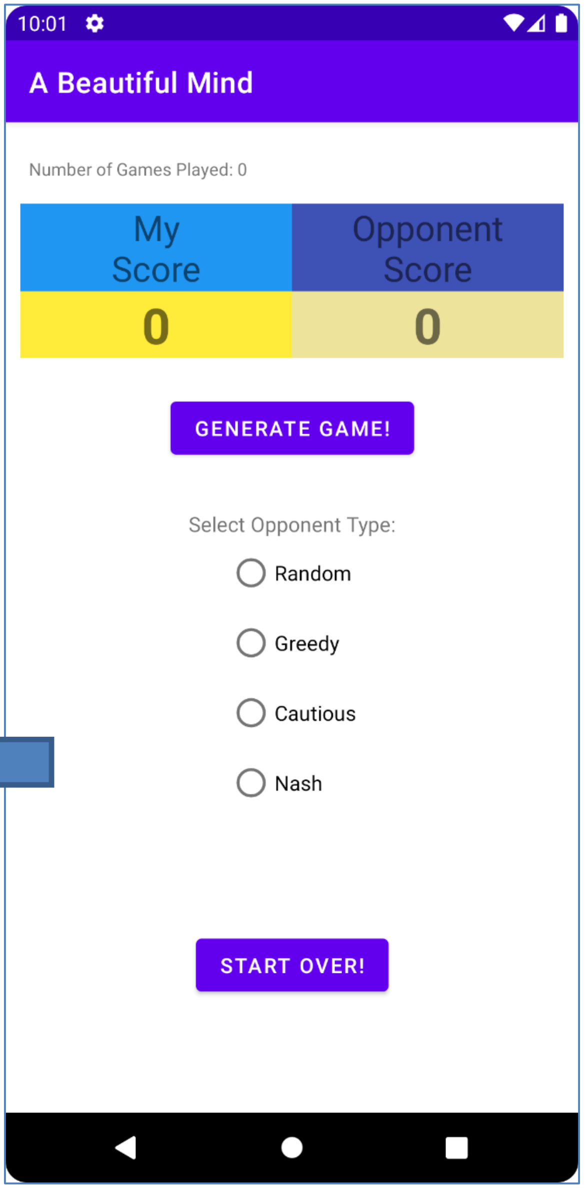
Now, it's your game ...

Your opponent plays according to this strategy: Random

	A	B
A	-3 / 3	-4 / 4
B	-2 / -3	1 / 0

Hint: All returns are given in the form x/y, where x specifies the reward for you, and y gives the reward for your opponent.
Hint: My action is specified by the row. The opponent's action is given by the column.

Result:
You chose: Action _
Opponent chose: Action _
You obtain: _ points
Opponent obtains: _ points



10:01

A Beautiful Mind

Number of Games Played: 0

My Score	Opponent Score
0	0

Select Opponent Type:
- ○ Random
- ○ Greedy
- ○ Cautious
- ○ Nash

Strategic Games (1)

- *Definition:* A strategic game G consists of
  - a finite set N (set of players or agents),
  - for each player $i \in N$ a non-empty action set A_i (the set of actions available to player i), with the total action set $A = \Pi_i A_i = A_1 \times A_2 \times \dots \times A_n$,
  - and for each player $i \in N$, a utility function $u_i: A \rightarrow R$ (utility, payoff, reward).
  - The game as a whole is thus a tuple with three components: $G = (N, (A_i), (u_i))$.
- Notes:
  - If A is a finite set, the game is said to be finite.
  - For simplicity, we will focus on N=2 and $A_1=A_2=\{a,b\}$.

Strategic Games (2)

- Process:
  - Each player i makes its decision about which action a_i to take.
  - All players execute their actions simultaneously.
  - This results in a joint action of the form: $a^* = (a_1, a_2, ..., a_n)$.
    - In our case, there are only four different joint actions: (a,a), (a,b), (b,a), (b,b)
  - Each player receives its payoff $u_i(a^*)$.
- Notes:
  - Naturally, each player will act rationally and try to maximize its own utility.
    ➔ But what is the right or necessary decision for this purpose?
  - While each player intends to maximize its own utility, the objective is not to intentionally harm the opponent.
    - It does not matter to Agent 1 what utility Agent 2 achieves (if one wanted to model this, the utility functions would have to be adjusted accordingly).

Strategic Games (3)

- Representation:
  - In 2-player games, a matrix is used.
  - Actions of player 1 are the rows.
  - Actions of player 2 are the columns.
  - Utility for each joint action is given as a pair (x, y).
  - Here, x represents the utility for player 1.
  - And y represents the utility for player 2.

	Player 2: Action L	Player 2: Action R
Player 1: Action O	x_{11},y_{11}	x_{12},y_{12}
Player 1: Action U	x_{21},y_{21}	x_{22},y_{22}

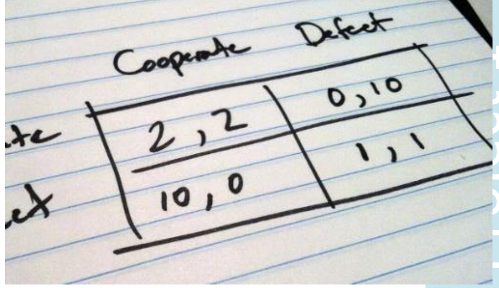
- Example:
  - Player 1 chooses action U, player 2 chooses action R.
  - Joint action (U, R) is executed.
  - Player 1 receives a utility of x_{22}, player 2 receives y_{22}.

Examples (1)

- Classical example: Strategic game “Battle of the sexes”
  - A man and a woman want to go out.
  - Both want to experience something together.
  - The man prefers a football stadium.
  - The woman prefers a concert.

	Musician: Action „Soccer“	**Musician: Action „Concert“**
Sportsman: Action „Soccer“	3,1	0,0
Sportsman: Action „Concert“	0,0	1,3

Examples (2)



- Example: Strategic game “Prisoner's Dilemma”
  - Two criminals are suspected of having committed a crime together.
  - Both are interrogated in separate rooms and have no opportunity to consult with each other or coordinate their behavior.
  - Maximum penalty for the crime: 6 years
  - If both prisoners decide to remain silent (defection), they will each be sentenced to 2 years in prison for minor offenses.
  - If both confess to the crime (cooperate), they will both receive a prison sentence, but because they cooperated with the authorities, they will not receive the maximum penalty, but only 4 years in prison.
  - If only one confesses (cooperate) and the other remains silent (defection), the first will receive a symbolic 1-year probationary sentence as a witness for the prosecution, and the other will receive the maximum sentence of 6 years.

	Prisoner 2: Action „defect“	**Prisoner 2: Action „cooperate“**
Prisoner 1: Action „defedt“	-2,-2	-6,-1
Prisoner 1: Action „cooperate“	-1,-6	-4,-4

Solutions to Strategic Games

- What does it mean to "solve" a strategic game?
  - It means to find the correct or best possible action for the considered agent.
  - Note: Instead of actions, we also speak of strategies!
  - Note: Instead of the overall actions of all agents, we also speak of a strategy profile!
- Quite different solution concepts are possible:
  - the elimination of strictly or weakly dominated strategies
  - heuristic approaches → Something we'll explore!
  - the finding of minimax strategies
    - to minimize the potential loss in zero-sum games,
  - the search for a Nash equilibrium state → Something we'll explore!
- Fundamental questions:
  - How difficult is it to find one of these solution methods?
  - Are there always solutions? → Something we'll explore!
  - If yes, are they uniquely determined? → Something we'll explore!

Elimination of Strictly Dominated Strategies (1)

- Notation:
  - Let $a = (a_1, a_2, \ldots, a_n)$ be a joint action.
  - Denote $a_{-i} = (a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n)$ as the joint action obtained by omitting the action of i.
  - Denote $(a_{-i}, a_i') = (a_1, \ldots, a_{i-1}, a_i', a_{i+1}, \ldots, a_n)$ as a new joint action where agent i chooses action a_i'.
- *Definition:* A strategy $a_j \in A$ of agent j is strictly dominated, if there exists a strategy a_j' such that for all joint actions $a \in A$:

$$u_j(a_{-j}, a_j') > u_j(a_{-j}, a_j)$$

- Note:
  - It is obvious that it would not be rational for agent j to play a strictly dominated strategy.
  - Because there are better strategies that promise greater utility for it.
  - Strict dominance exists only in the rarest cases.

Elimination of Strictly Dominated Strategies (2)

- Iterative elimination of strictly dominated strategies:
  - Since it is never advisable to play according to a strictly dominated strategy, it can be eliminated from the game.
  - This can be done iteratively.
  - If one arrives at a single remaining joint action in this way, the result is unique.
  - This result can be considered the “outcome of the game” because it represents the only rational outcome for the players.

Elimination of Strictly Dominated Strategies (3)

- Example: Prisoner's Dilemma
  - Prisoner 1 decides that “cooperate” is the better choice for him because:
    - If Prisoner 2 chooses “defect”, then 1 year in prison is better than 2 years.
    - If Prisoner 2 chooses “cooperate”, then 4 years in prison is better than 6 years.
  - Prisoner 2 realizes that Prisoner 1 excludes the option of “defect”.
  - Therefore, he also excludes the option of “defect” for himself because 4 years in prison are better than 6 years.
- Result: Both cooperate!

	Prisoner 2: Action „defect“	**Prisoner 2: Action „cooperate“**
Prisoner 1: Action „defect“	~~-2,-2~~	~~-6,-1~~
Prisoner 1: Action „cooperate“	~~-1,-6~~	-4,-4

Elimination of Strictly Dominated Strategies (4)

- Example:
  - Who dominates whom?
    - b3 dominates b4 → eliminate b4
    - a1 dominates a4 → eliminate a4
    - b2 dominates b3 → eliminate b3
    - a2 dominates a1 → eliminate a1
    - b2 dominates b1 → eliminate b1
    - a2 dominates a3 → eliminate a3
  - Result: (a2, b2)

	b1	b2	b3	b4
a1	~~(1,7)~~	~~(2,5)~~	~~(7,2)~~	~~(0,1)~~
a2	~~(5,2)~~	(3,3)	~~(5,2)~~	~~(0,1)~~
a3	~~(7,0)~~	~~(2,5)~~	~~(0,4)~~	~~(0,2)~~
a4	~~(0,0)~~	~~(1,-2)~~	~~(5,0)~~	~~(9,-1)~~

Elimination of Weakly Dominated Strategies (1)

- Instead of demanding strict dominance, one can also settle for weak dominance.
- *Definition:* A strategy $a_j^* \in A_j$ of agent j is weakly dominated if there exists a strategy a_j', such that for all joint actions $a \in A$:

  $$u_j(a_{-j}, a_j') \geq u_j(a_{-j}, a_j)$$

  and, additionally, for at least one joint action $a \in A$:

  $$u_j(a_{-j}, a_j') > u_j(a_{-j}, a_j^*)$$
- Note:
  - The result of such an elimination procedure is not necessarily unique.

Elimination of Weakly Dominated Strategies (2)

- Example:
  - M weakly dominates U → eliminate U
    - because 2>0 and 1≥1
  - L weakly dominates R → eliminate R
    - because 1>0 and 1≥1

	L	R
O	2,1	~~0,0~~
M	2,1	~~1,1~~
U	~~0,0~~	~~1,1~~

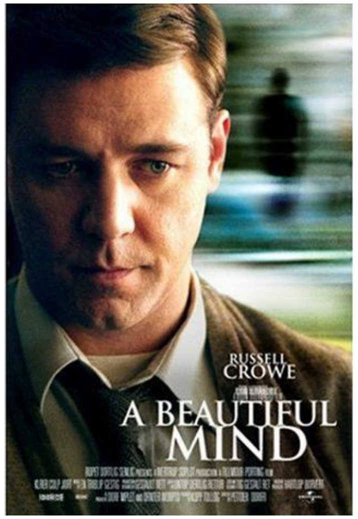
	L	R
O	~~2,1~~	~~0,0~~
M	~~2,1~~	1,1
U	~~0,0~~	1,1

  - M weakly dominates O → eliminate O
    - because 2≥2 und 1>0
  - R weakly dominates L → eliminate L
    - because 1≥1 und 1 >0

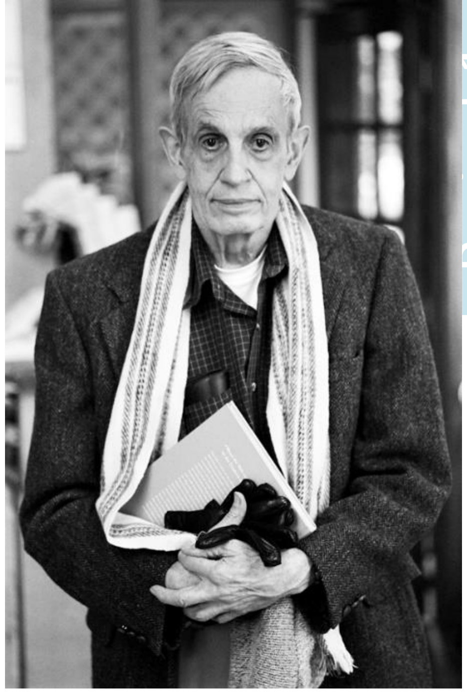
Nash Equlibria (1)

- Dominant strategies are an attractive solution concept for strategic games.
  But what to do, if there are no dominated strategies?
  - Fact: Dominant strategies are rare in practice.
- Example: Strategic game “Dove and Hawk”
  - Distribution of a (to be divided) loot
  - Behave like a dove or like a hawk!
  - Doves share fairly, hawks only leave little for doves, two hawks fight and harm each other.
    ➔ There are no dominated strategies (neither strictly nor weakly).
- Alternative solution concept:
  Nash Equilibrium
  (named after Nobel laureate John Nash, 1951)

	Dove	Hawk
Dove	(3,3)	(1,4)
Hawk	(4,1)	(0,0)



Nash Equlibria (2)



- *Definition:* A **Nash Equilibrium** is joint action (aka strategy profile) $a^* \epsilon A$ such that it holds for each player i:

  $$u_i(a^*) = u_i(a_{-i}^*, a_i^*) \geq u_i(a_{-i}^*, a_i) \text{ for all } a_i \epsilon A_i$$

- Explanations:
  - It is a joint action in which no player has any reason to deviate from its individual action.
  - Choosing a different action would mean a deterioration for each player.
  - As a solution concept, it is not as convincing as joint actions resulting from the iterative elimination of dominated strategies, but still a "reasonable" solution.
  - If there is a unique solution in the iterative elimination of strictly dominated strategies, then this solution is always a Nash equilibrium.

Nash Equlibria (3)

- Example: Strategic game „Prisoner‘s Dilemma“
  - Which joint actions are Nash equilibria?
  - „defect-defect “ → no NE
  - „defect-cooperate “ → no NE
  - „cooperate-defect “ → no NE
  - „cooperate-cooperate “ → **Nash Equlibrium!**

	Prisoner 2: Action „defect“	**Prisoner 2: Action „cooperate“**
Prisoner 1: Action „defect“	-2,-2	-6,-1
Prisoner 1: Action „cooperate“	-1,-6	-4,-4

Nash Equlibria (4)

- Example: Strategic game „Dove and Hawk“
  - „Dove-Dove“ → no NE
  - „Hawk-Hawk“ → no NE
  - „Dove-Hawk“ → NE!
  - „Hawk-Dove“ → NE!

	Dove	Hawk
Dove	(3,3)	(1,4)
Hawk	(4,1)	(0,0)

- Insight: Nash equilibria in strategic games are not necessarily uniquely determined.
  - There may be multiple NEs (as well as a single one or none at all).

Nash Equlibria (5)

- Example: Strategic game „Matching Pennies“
  - “two-action version” of “Rock, Paper, Scissors”
  - Two players, each secretly flips a coin to “heads” or “tails”.
  - Agreement:
    - both chose “heads” → Player A receives both coins
    - both chose “tails” → Player A receives both coins
  - No agreement:
    - If both coins show different results → Player B receives both coins
  - The game is “strictly competitive”!
  - Question: Is there an NE?
    - No, there is no Nash equilibrium in this game!
    - What to do? → Usage of randomized strategies

	Spieler B: Aktion „wirft Kopf“	**Spieler B: Aktion „wirft Zahl“**
Spieler A: Aktion „wirft Kopf“	+1,-1	-1,+1
Spieler A: Aktion „wirft Zahl“	-1,+1	+1,-1

Laboratory – App “Project 1”

- Informal description of Project 1 „A Beautiful Mind“:
  - The app shall realize a simple game in which the user plays strategic games against the computer. The computer opponent shall be capable of playing against one out of four different strategies: random, greedy, cautious, and Nash. In the app‘s first activity (main activity), the overall score achieved so far shall be displayed (score board). Also, in the first activity, the user may select one these four computer strategies. By clicking a button, a second activity (game activity) shall be started. The information about the selected opponent strategy shall be transferred from the main activity to the game activity. Upon its creation, the game activity shall create a random two-player stochastic game that allows each player two actions ($A_{human}=A_{computer}=\{a,b\}$). The utilities (also called game points) of the game shall be determined randomly from the range [-5,5] and displayed in a game matrix where the human‘s actions are denoted in the rows, and the computer opponent‘s actions are displayed in the game matrix‘ columns. Also, the computer opponent‘s strategy shall be mentioned above the table. A text information below the table shall briefly explain the structure of the table to the user. The game activity shall have two buttons that are placed below the table and which the human player must use to pick either action a or b. After having picked one of these actions, (a) both buttons shall be inactivated, (b) the computer player‘s action shall be determined, (c) the corresponding entry‘s background in the game matrix shall be colored, and (d) the selected actions as well as the resulting utilities shall be displayed below the buttons. At the bottom of the game activity‘s screen, there shall be another button (dismiss button). [continued on the next slide …]

Laboratory – App “Project 1”

- Informal description of Project 1 „A Beautiful Mind“:
  - When clicking this button without the human player having selected one of the two actions before, the game shall be discarded and the game activity shall finish, where some information shall be displayed that the recent game has been aborted, when returning to the main activity. When clicking the dismiss button after the human player has selected one of the two actions, the game activity shall be finished and return the utilities for the human and the computer player to the main activity. Upon returning to the main activity, the score board shall be updated accordingly. The main activity shall display the number of games played so far at its top. The main activity shall have a button to start over, which resets the score board and the counter of games played so far. When reselecting the opponent strategy in the main activity, the scores and number of games played shall be reset to zero, too. When selecting the opponent strategy Nash in the main activity, an image of John Nash shall be displayed near to the selection. That image shall disappear when the user selects a strategy different from Nash.
- In the context of this course, you are permitted to utilize an image of John Nash from Wikipedia (cf. https://de.wikipedia.org/wiki/John_Forbes_Nash_Jr.) or, alternatively, the film ad photo of the belonging Hollywood movie „A Beautiful Mind“ (e.g. https://www.imdb.com/title/tt0268978/mediaviewer/rm928937216/).

Laboratory – App “Project 1”

- Definition of the four different strategies according to which the computer selects its action:
  - Note: In the following „1“ is the subscript for the human player, and „2“ is the subscript for the computer player.
  - Random: $a_2 = a \; or \; b \; each \; with \; probability \; 0.5$
  - Greedy: $a_2 = argmax_{y \in \{a,b\}} \sum_{x \in \{a,b\}} u_2(x,y)$
  - Cautious: $a_2 = argmin_{y \in \{a,b\}} \sum_{x \in \{a,b\}} u_1(x,y)$
  - Nash: $a_2 = a_2^*$ where $a^* = (a_1^*, a_2^*)$ is the unique Nash equilibrium of the strategic game. If no or multiple Nash equlibria exist in the game, then no action a_2 is selected.

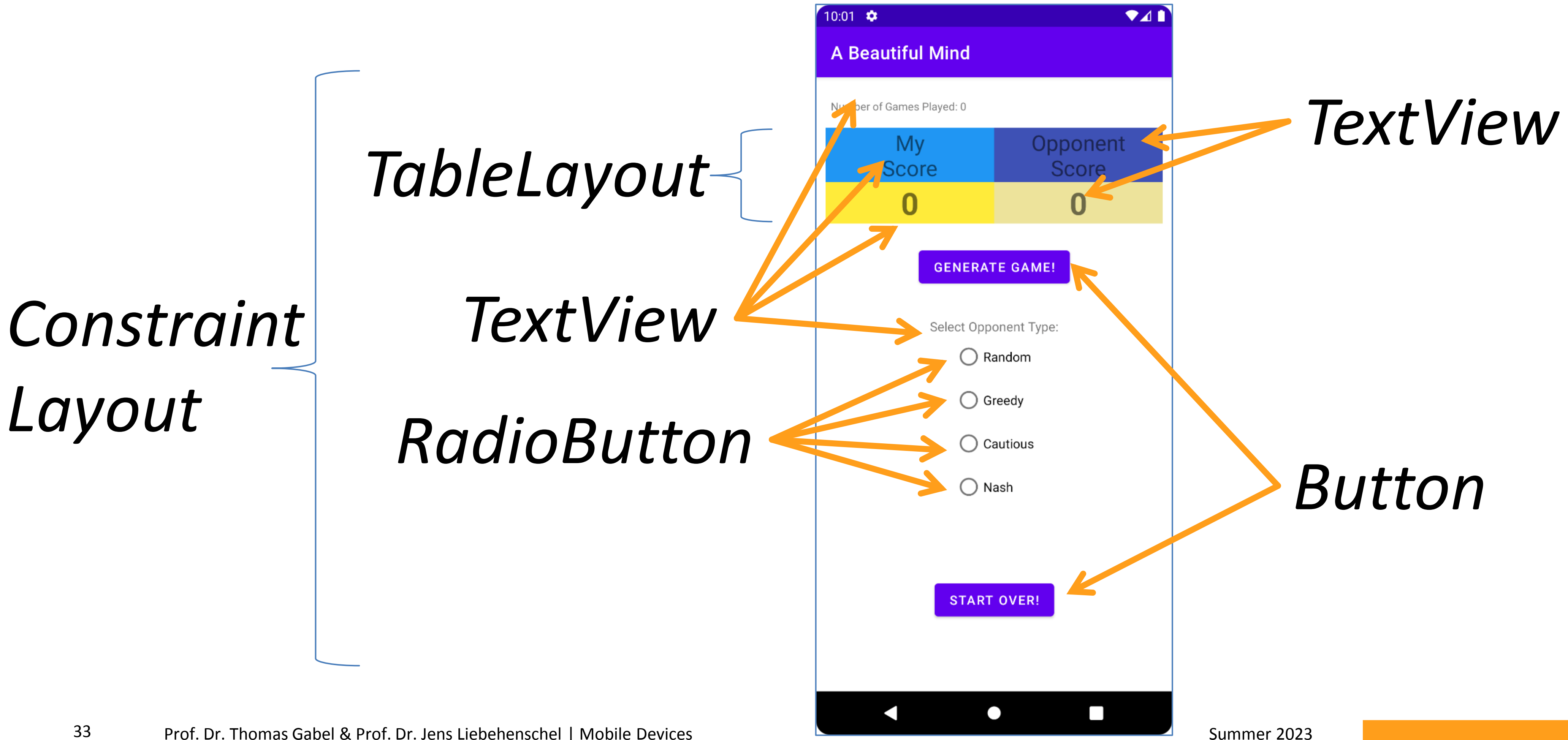
Step 1: How to Start?

Update as of May‘23:
If you use the latest Android Studio Version „Flamingo 22.2.1 Patch 1“, there has been a renaming of Project Templates! In particular, „Empty Activity“ has been renamed to „Emtpy View Activity“, which you are required to use. This implies the usage of XML-based layout definitons (cf. slides on Lecture Unit 6).

- Create an Android app named “A Beautiful Mind”
  - Minimum SDK: API 24 (Android 7.0)
  - Target and Compile SDK: API 33 (Android 13)
  - New Project → Empty Activity → Name “A Beautiful Mind”
  - The app’s package name must be `de.fra_uas.fb2.mobiledevices.abeautifulmind`
  - Programming language: Kotlin or Java
- Look at the files in the project browser
- Create a virtual device *Pixel 3a* in Android Studio
- Run the app in this virtual device

Step 2: Layout the Main Screen

- The screen layout shall look as given on next slide
  - Use the graphical and the textual view during design

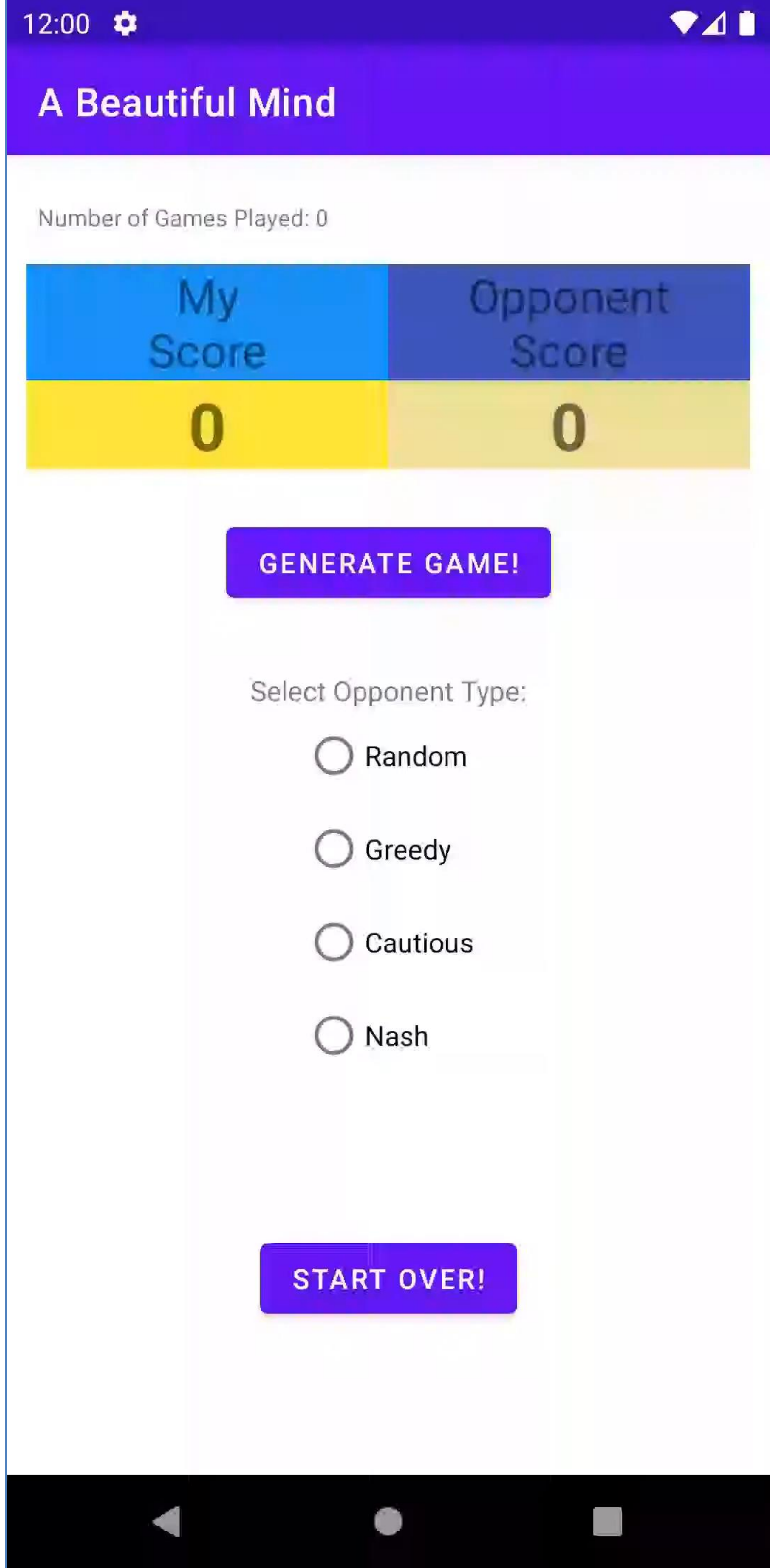


Step 2 - Generalize & Test

- Use string resources for every text!
- Preview different screen sizes, e.g.
  - 1080 x 2220 px
  - 720 x 1280 px
- Use *ConstraintLayout* (and, if needed, *LinearLayout*) for grouping user interface elements
- Test the app on (by playing around with the UI elements)
  - the virtual device *Pixel 3a* (and e.g. a virtual *Nexus 5X*)
  - a physical device (if available)

Step 2 - Preview

- Video (screen capture) showing the final expected functioning of the final app.
- Note:
  - You don‘t have to implemented all at once. ;-)
  - The video does not show all required features.
    → see next slides


12:00

A Beautiful Mind

Number of Games Played: 0

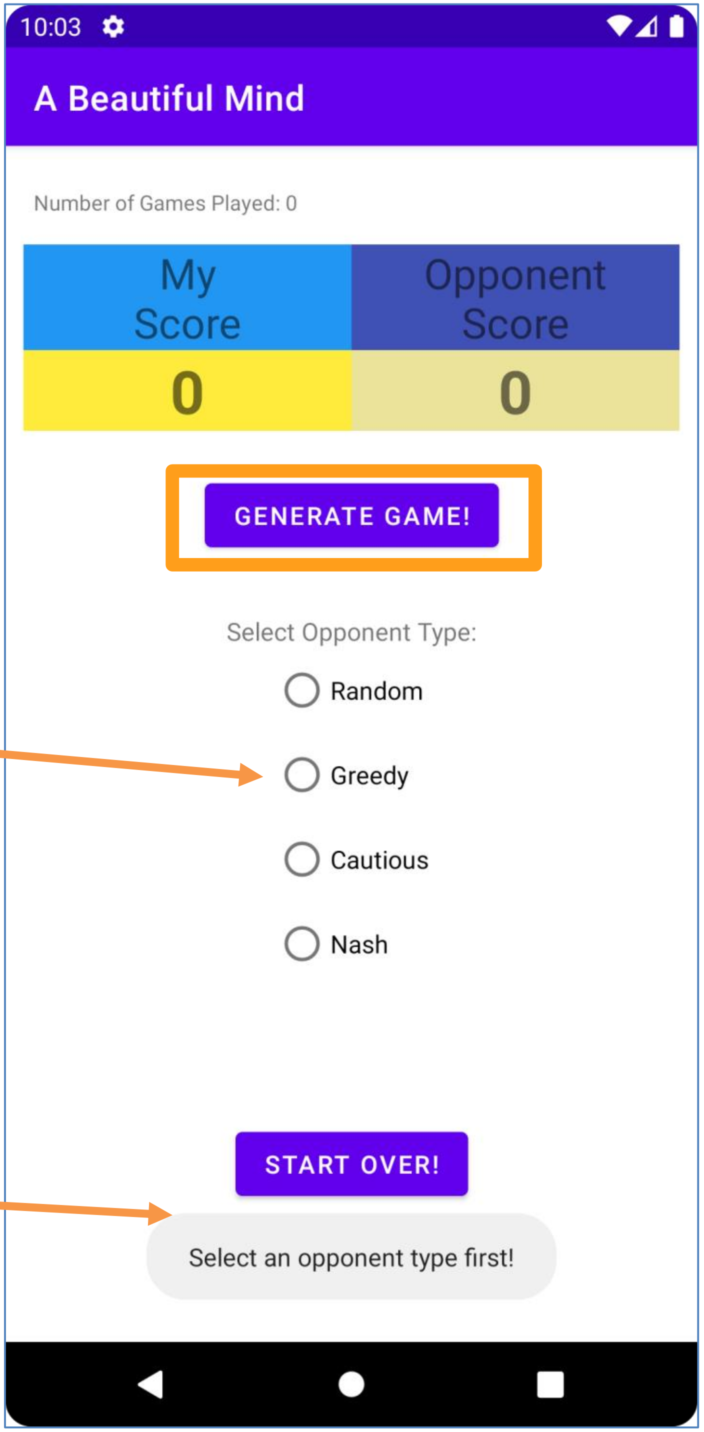
My Score	Opponent Score
0	0

Step 3 - Add Functionality

- Add functionality to the app according to the description on the next slides
- Continue to test the app on
  - the virtual device *Pixel 3a*
  - a physical device (if available)
- Debug the app on the virtual device *Pixel 3a*
  - Use breakpoints
  - Look at variables at runtime

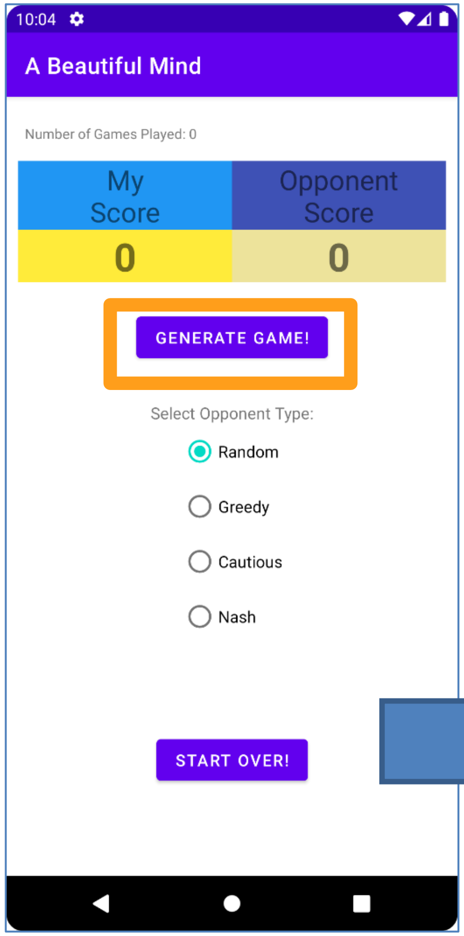
Step 3

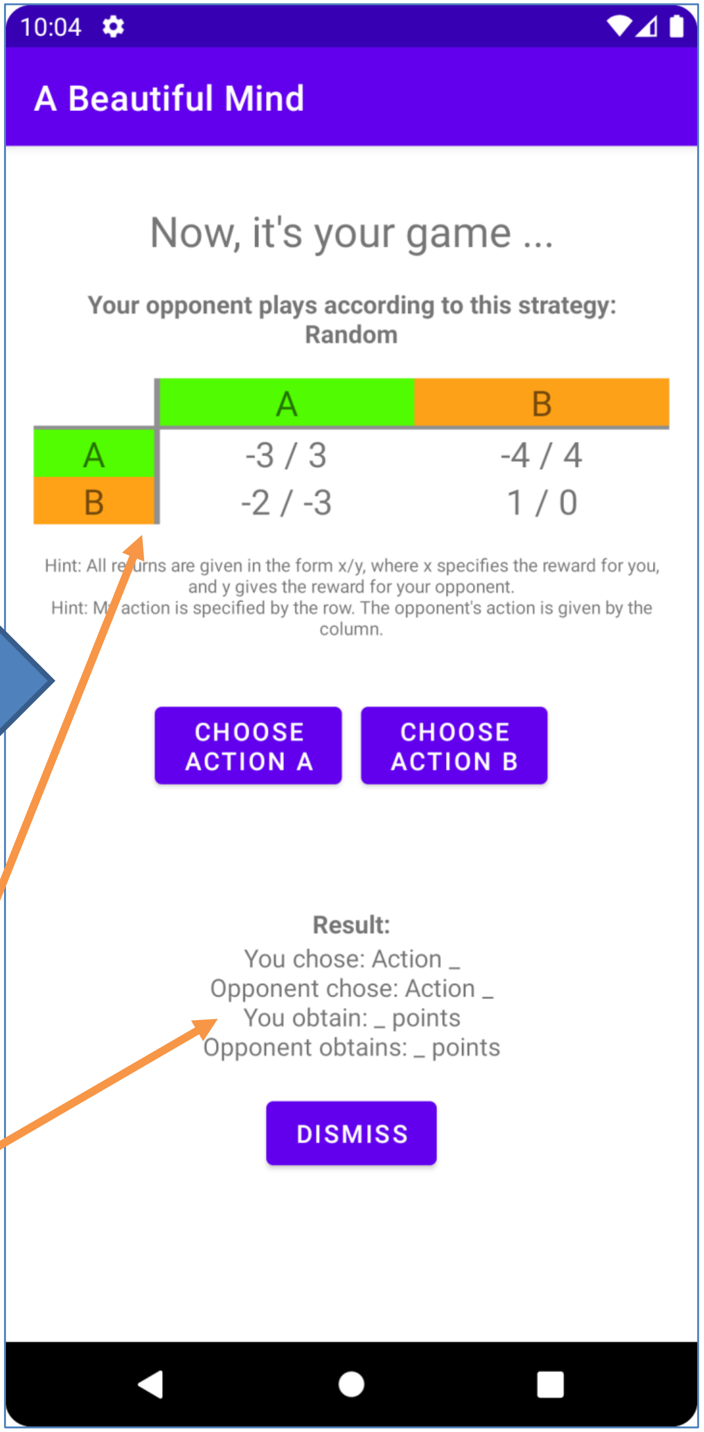
- Number of games played
  → initially zero
- RadioGroup with four options for opponent types
- Clicking the generate button without having specified an opponent type (strategy)
  → Toast notification!



Step 3

- Clicking „Generate Game“ brings the user to the game activity where the selected opponent strategy is handed over
  - e.g. Random
- Random game matrix is generated (all utilities from [-5,5]) & displayed
  - human action in rows, computer in columns
  - result initially with blanks „_“




10:04

A Beautiful Mind

Now, it's your game ...

Your opponent plays according to this strategy: Random

	A	B
A	-3 / 3	-4 / 4
B	-2 / -3	1 / 0

Hint: All returns are given in the form x/y, where x specifies the reward for you, and y gives the reward for your opponent.
Hint: My action is specified by the row. The opponent's action is given by the column.

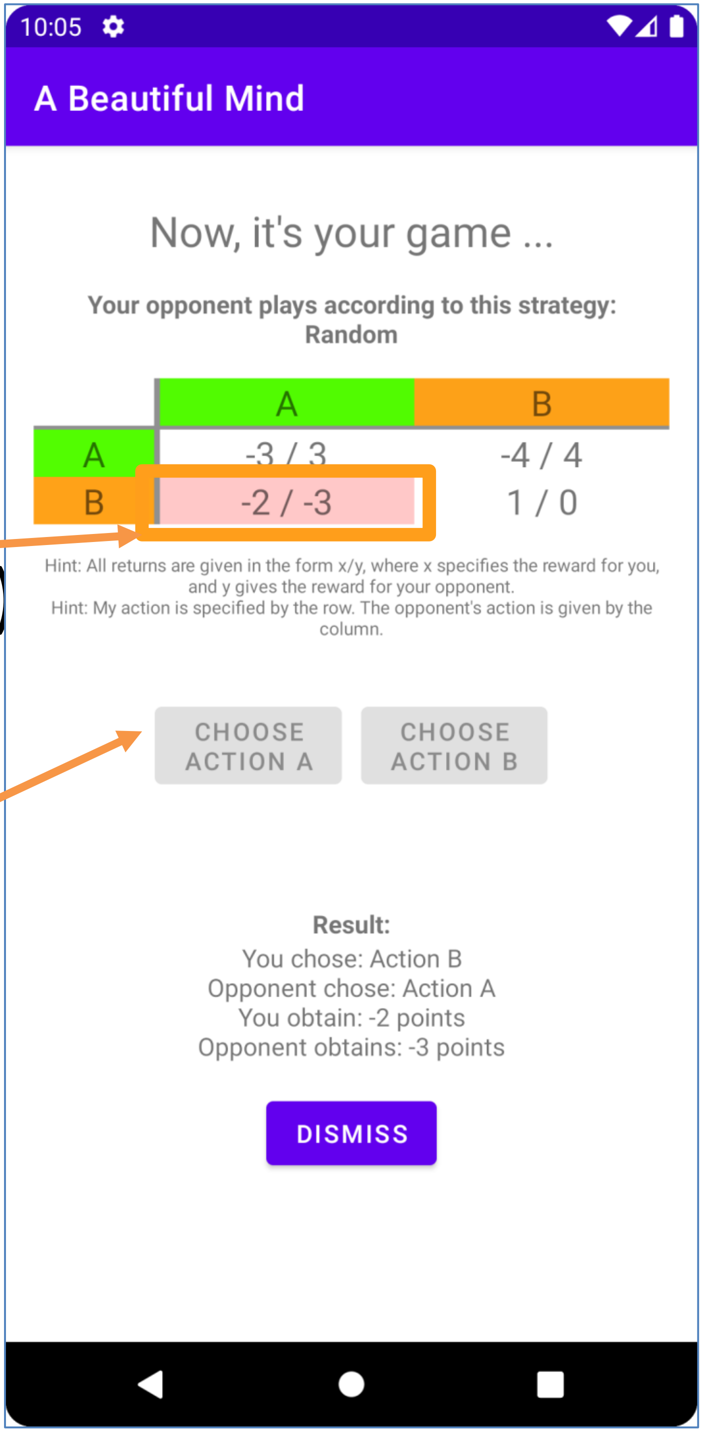
CHOOSE ACTION A   CHOOSE ACTION B

Result:
You chose: Action _
Opponent chose: Action _
You obtain: _ points
Opponent obtains: _ points

DISMISS

Step 3

- User has clicked on „Chose Action A“
  - result in matrix is highlighted (A,B)
    - – computer selected (randomly) action B
  - both action choice buttons become disabled



Now, it's your game ...

Your opponent plays according to this strategy: Random

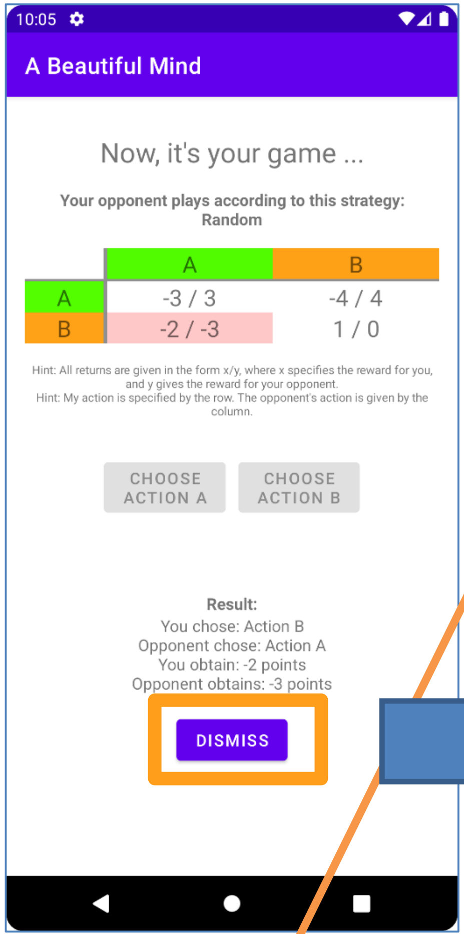
	A	B
A	-3 / 3	-4 / 4
B	-2 / -3	1 / 0

Hint: All returns are given in the form x/y, where x specifies the reward for you, and y gives the reward for your opponent.
Hint: My action is specified by the row. The opponent's action is given by the column.

Result:
You chose: Action B
Opponent chose: Action A
You obtain: -2 points
Opponent obtains: -3 points

Step 3

- User finished by clicking „Dismiss“
  - result (-2/-3) is returned
- Back in the main activity
  - number of games played gets incremented
  - score board gets updated



10:05

A Beautiful Mind

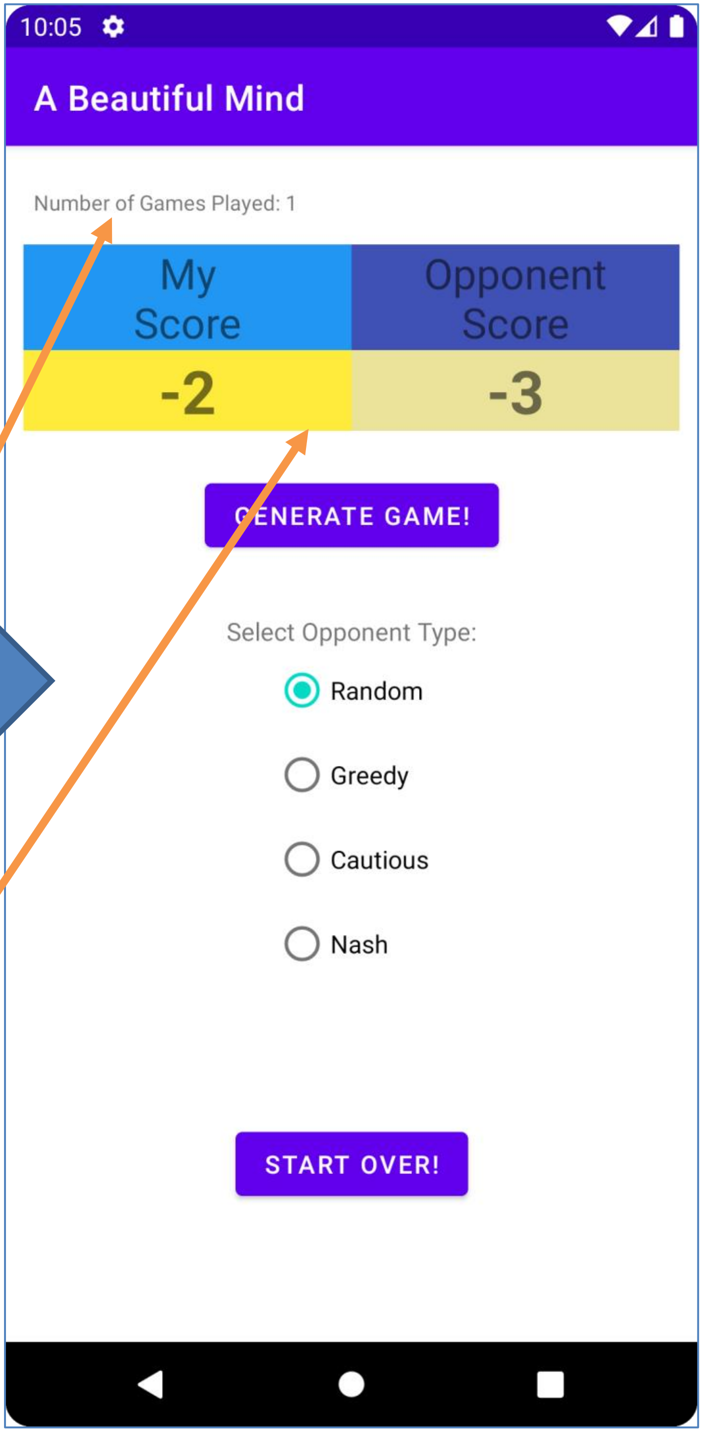
Now, it's your game ...

Your opponent plays according to this strategy: Random

	A	B
A	-3 / 3	-4 / 4
B	-2 / -3	1 / 0

Hint: All returns are given in the form x/y, where x specifies the reward for you, and y gives the reward for your opponent.
Hint: My action is specified by the row. The opponent's action is given by the column.

Result:
You chose: Action B
Opponent chose: Action A
You obtain: -2 points
Opponent obtains: -3 points



10:05

A Beautiful Mind

Number of Games Played: 1

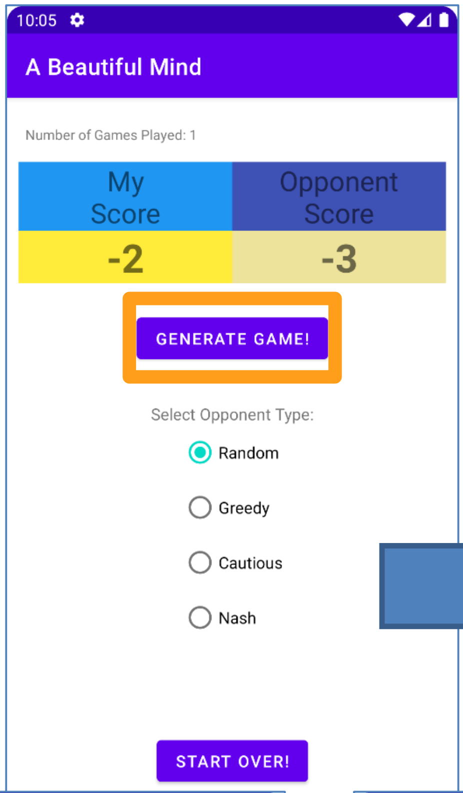
My Score	Opponent Score
-2	-3

Select Opponent Type:

[✓ Random] Greedy Cautious Nash

Step 3

- Another game is started by clicking „Generate Game“
  - randomly different game matrix
- User chooses action A, computer also A
  - result of (3/-2)
- After „Dismiss“, the app returns to the main activity
  - integrates the result



10:05

A Beautiful Mind

Number of Games Played: 1

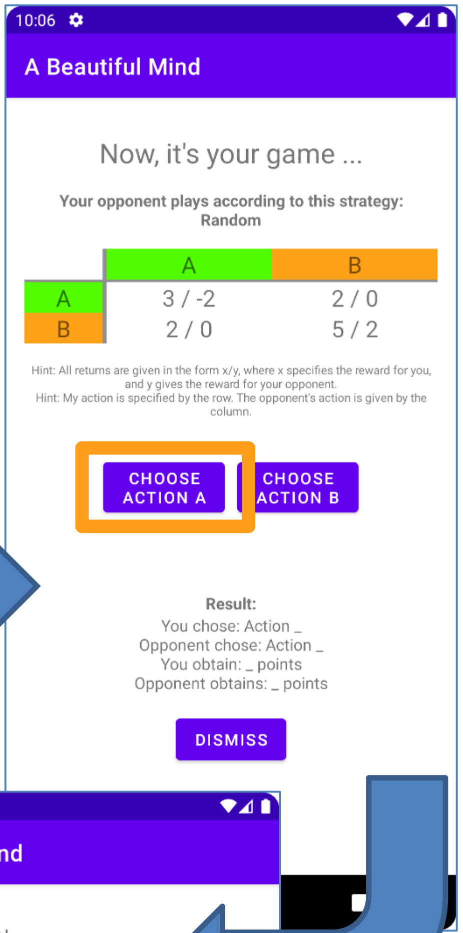
My Score	Opponent Score
-2	-3

GENERATE GAME!

Select Opponent Type:

- [✓ Random]
- Greedy
- Cautious
- Nash

START OVER!



10:06

A Beautiful Mind

Now, it's your game ...

Your opponent plays according to this strategy: Random

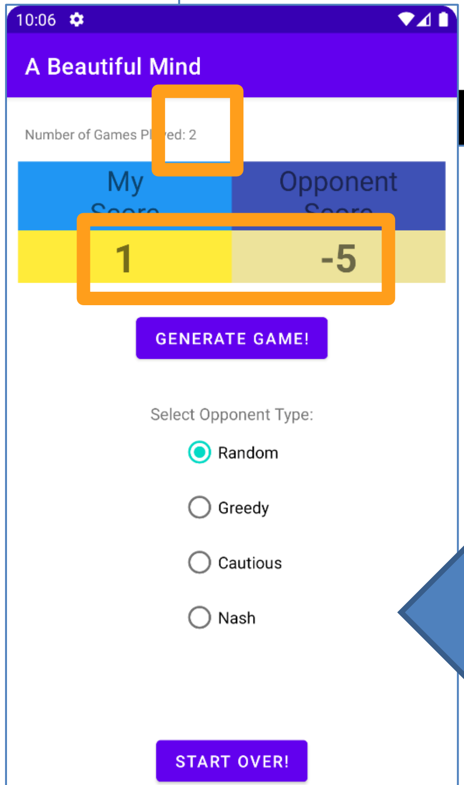
	A	B
A	3 / -2	2 / 0
B	2 / 0	5 / 2

Hint: All returns are given in the form x/y, where x specifies the reward for you, and y gives the reward for your opponent.
Hint: My action is specified by the row. The opponent's action is given by the column.

CHOOSE ACTION A | CHOOSE ACTION B

Result:
You chose: Action _
Opponent chose: Action _
You obtain: _ points
Opponent obtains: _ points

DISMISS



10:06

A Beautiful Mind

Number of Games Played: 2

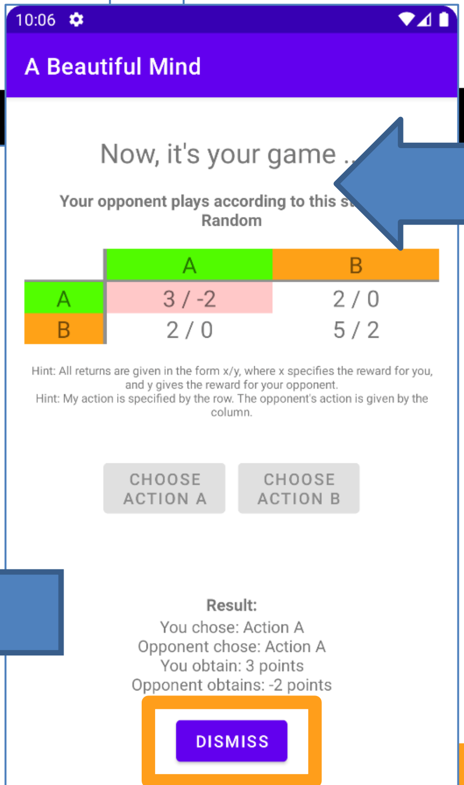
My Score	Opponent Score
1	-5

GENERATE GAME!

Select Opponent Type:

- [✓ Random]
- Greedy
- Cautious
- Nash

START OVER!



10:06

A Beautiful Mind

Now, it's your game ...

Your opponent plays according to this strategy: Random

	A	B
A	3 / -2	2 / 0
B	2 / 0	5 / 2

Hint: All returns are given in the form x/y, where x specifies the reward for you, and y gives the reward for your opponent.
Hint: My action is specified by the row. The opponent's action is given by the column.

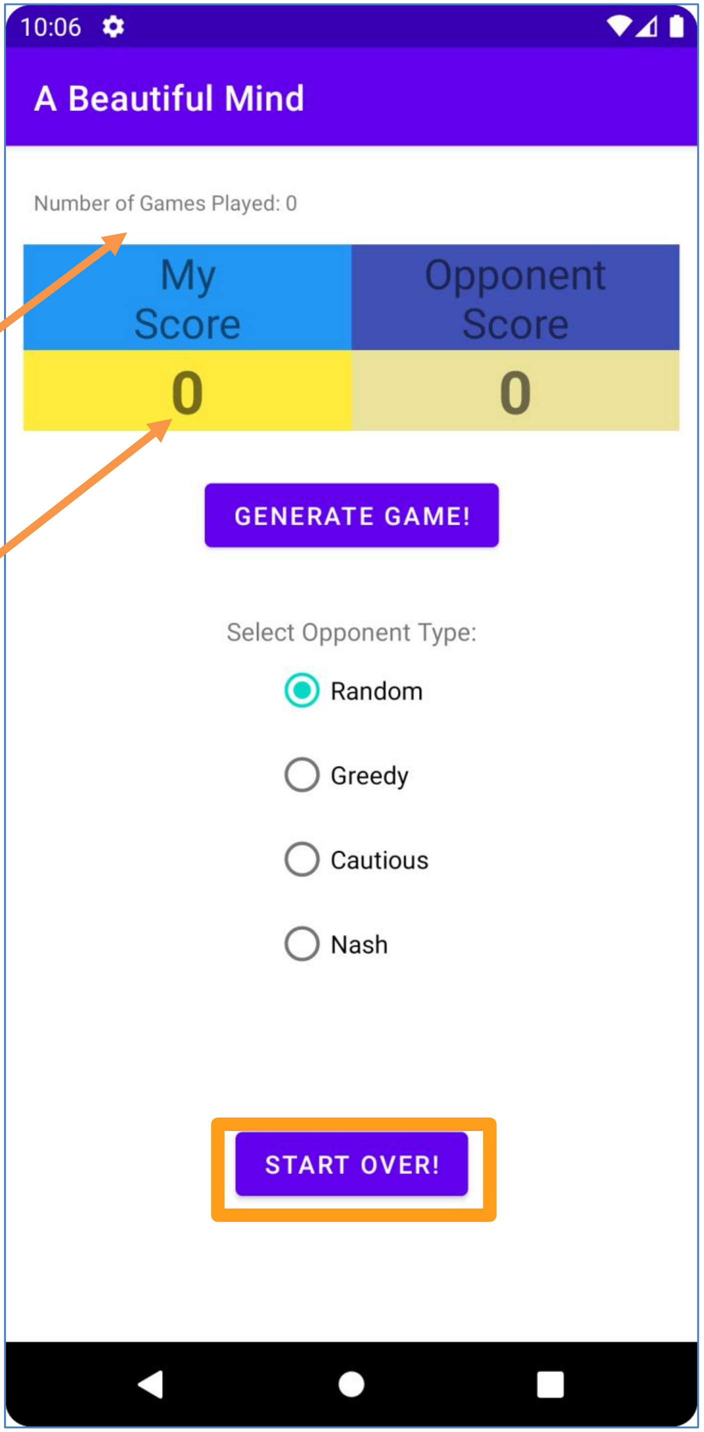
CHOOSE ACTION A | CHOOSE ACTION B

Result:
You chose: Action A
Opponent chose: Action A
You obtain: 3 points
Opponent obtains: -2 points

DISMISS

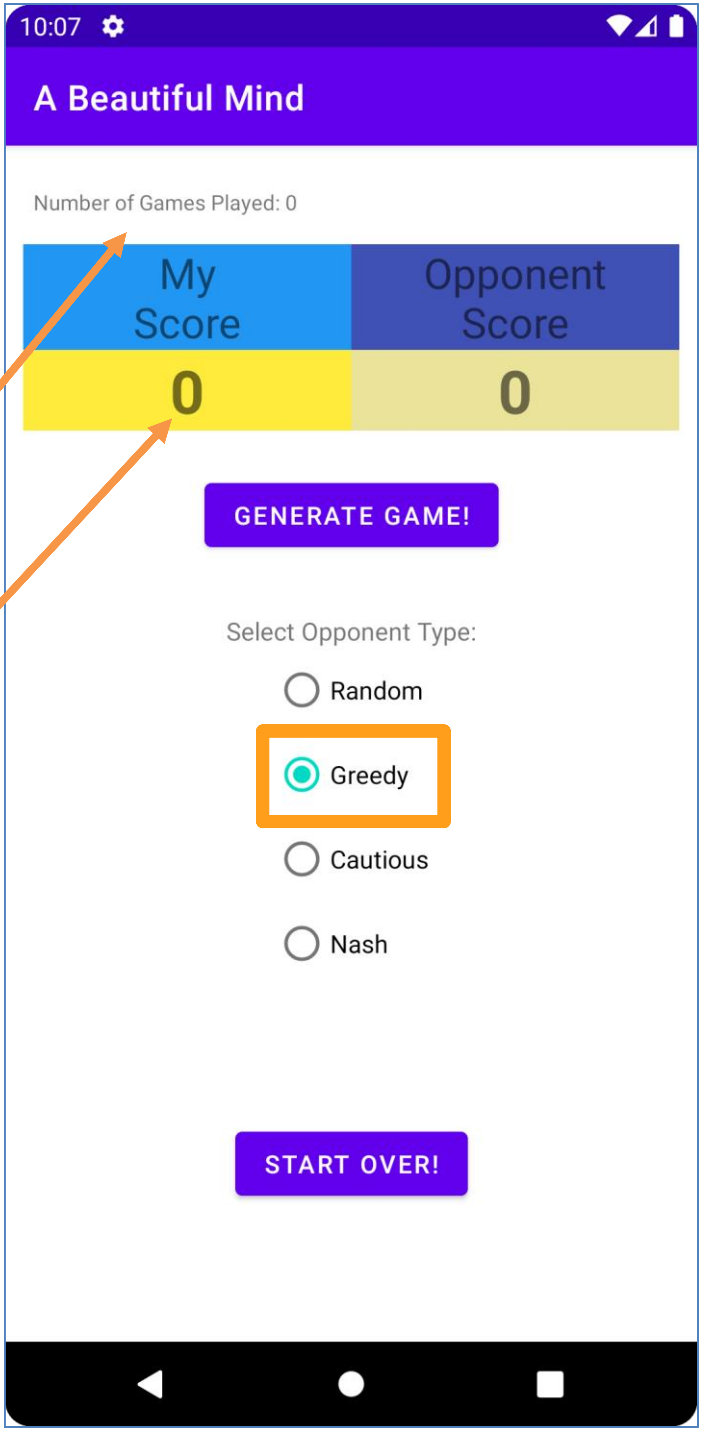
Step 3

- By clicking „Start Over“ …
  - the number of games played are resetted
  - the current score is resetted



Step 3

- By selecting another opponent type (here: Greedy), also
  - the number of games played are resetted
  - the current score is resetted


10:07

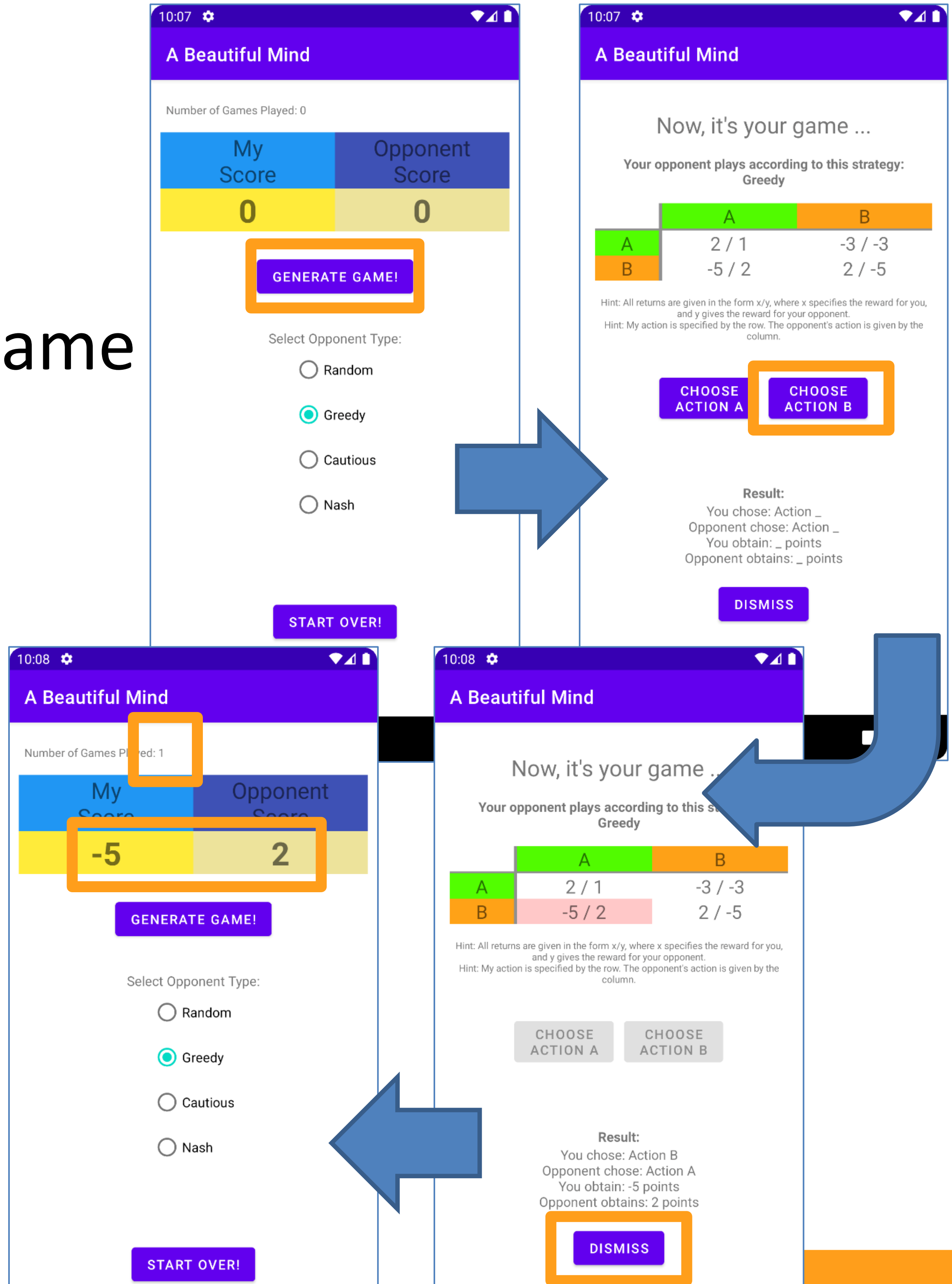
A Beautiful Mind

Number of Games Played: 0

My Score	Opponent Score
0	0

Step 3

- Example run for a game against a „Greedy“ opponent



10:07
A Beautiful Mind
Number of Games Played: 0

My Score	Opponent Score
0	0

GENERATE GAME!
Select Opponent Type:
- Random
- [✓ Greedy]
- Cautious
- Nash

START OVER!

10:07
A Beautiful Mind
Now, it's your game ...
Your opponent plays according to this strategy: Greedy

	A	B
A	2 / 1	-3 / -3
B	-5 / 2	2 / -5

Hint: All returns are given in the form x/y, where x specifies the reward for you, and y gives the reward for your opponent.
Hint: My action is specified by the row. The opponent's action is given by the column.

CHOOSE ACTION A   CHOOSE ACTION B

Result:
You chose: Action _
Opponent chose: Action _
You obtain: _ points
Opponent obtains: _ points

DISMISS

10:08
A Beautiful Mind
Now, it's your game ...
Your opponent plays according to this s
Greedy

	A	B
A	2 / 1	-3 / -3
B	-5 / 2	2 / -5

Hint: All returns are given in the form x/y, where x specifies the reward for you, and y gives the reward for your opponent.
Hint: My action is specified by the row. The opponent's action is given by the column.

CHOOSE ACTION A   CHOOSE ACTION B

Result:
You chose: Action B
Opponent chose: Action A
You obtain: -5 points
Opponent obtains: 2 points

DISMISS

10:08
A Beautiful Mind
Number of Games Played: 1

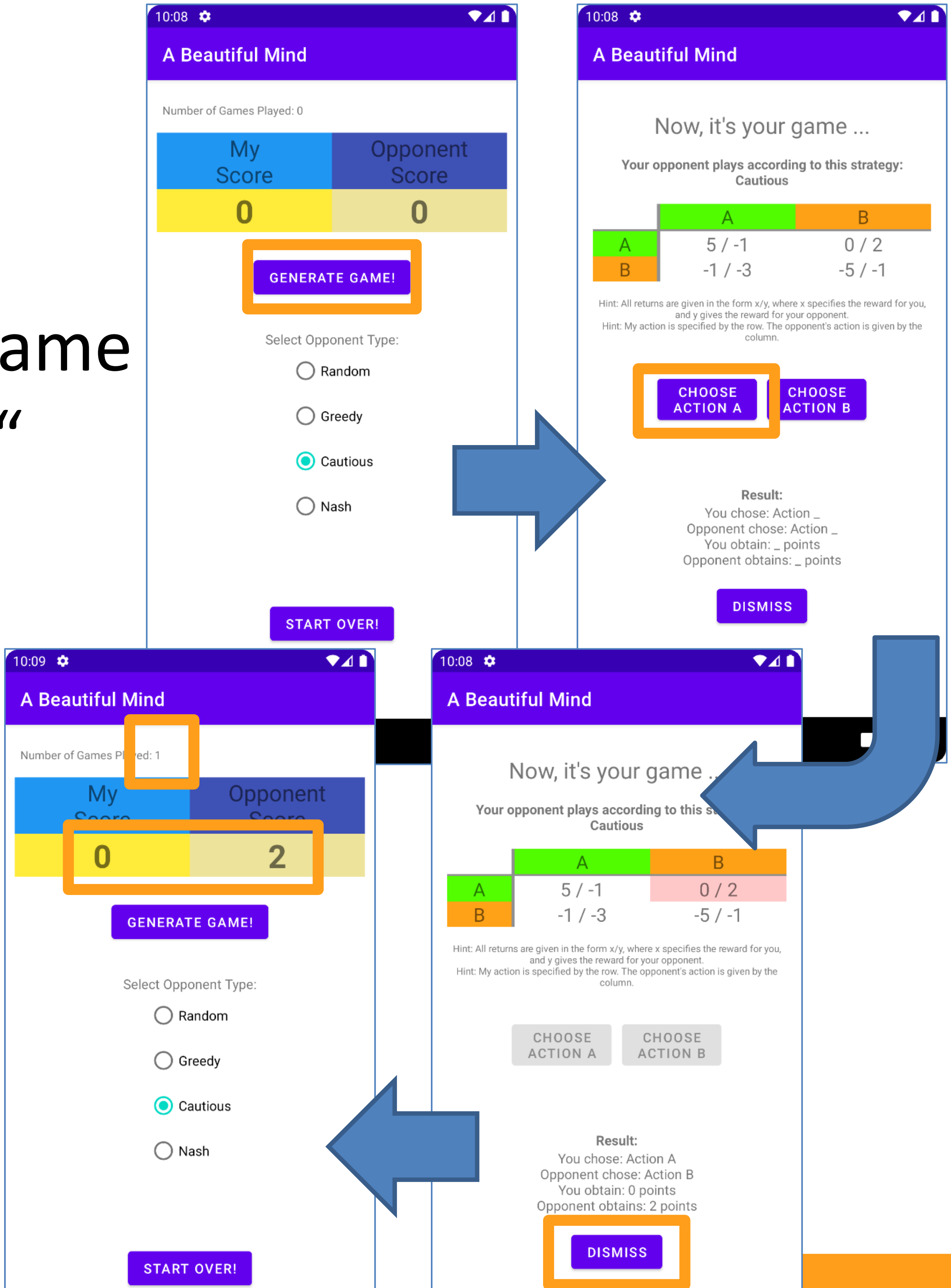
My Score	Opponent Score
-5	2

GENERATE GAME!
Select Opponent Type:
- Random
- [✓ Greedy]
- Cautious
- Nash

START OVER!

Step 3

- Example run for a game against a „Cautious“ opponent



10:08

A Beautiful Mind

Number of Games Played: 0

My Score	Opponent Score
0	0

GENERATE GAME!

Select Opponent Type:

- Random
- Greedy
- [✓ Cautious]
- Nash

START OVER!

10:08

A Beautiful Mind

Now, it's your game ...

Your opponent plays according to this strategy: Cautious

	A	B
A	5 / -1	0 / 2
B	-1 / -3	-5 / -1

Hint: All returns are given in the form x/y, where x specifies the reward for you, and y gives the reward for your opponent.
Hint: My action is specified by the row. The opponent's action is given by the column.

CHOOSE ACTION A | CHOOSE ACTION B

Result:
You chose: Action _
Opponent chose: Action _
You obtain: _ points
Opponent obtains: _ points

DISMISS

10:08

A Beautiful Mind

Now, it's your game ...

Your opponent plays according to this s Cautious

	A	B
A	5 / -1	0 / 2
B	-1 / -3	-5 / -1

Hint: All returns are given in the form x/y, where x specifies the reward for you, and y gives the reward for your opponent.
Hint: My action is specified by the row. The opponent's action is given by the column.

CHOOSE ACTION A | CHOOSE ACTION B

Result:
You chose: Action A
Opponent chose: Action B
You obtain: 0 points
Opponent obtains: 2 points

DISMISS

10:09

A Beautiful Mind

Number of Games Pl ed: 1

My Score	Opponent Score
0	2

GENERATE GAME!

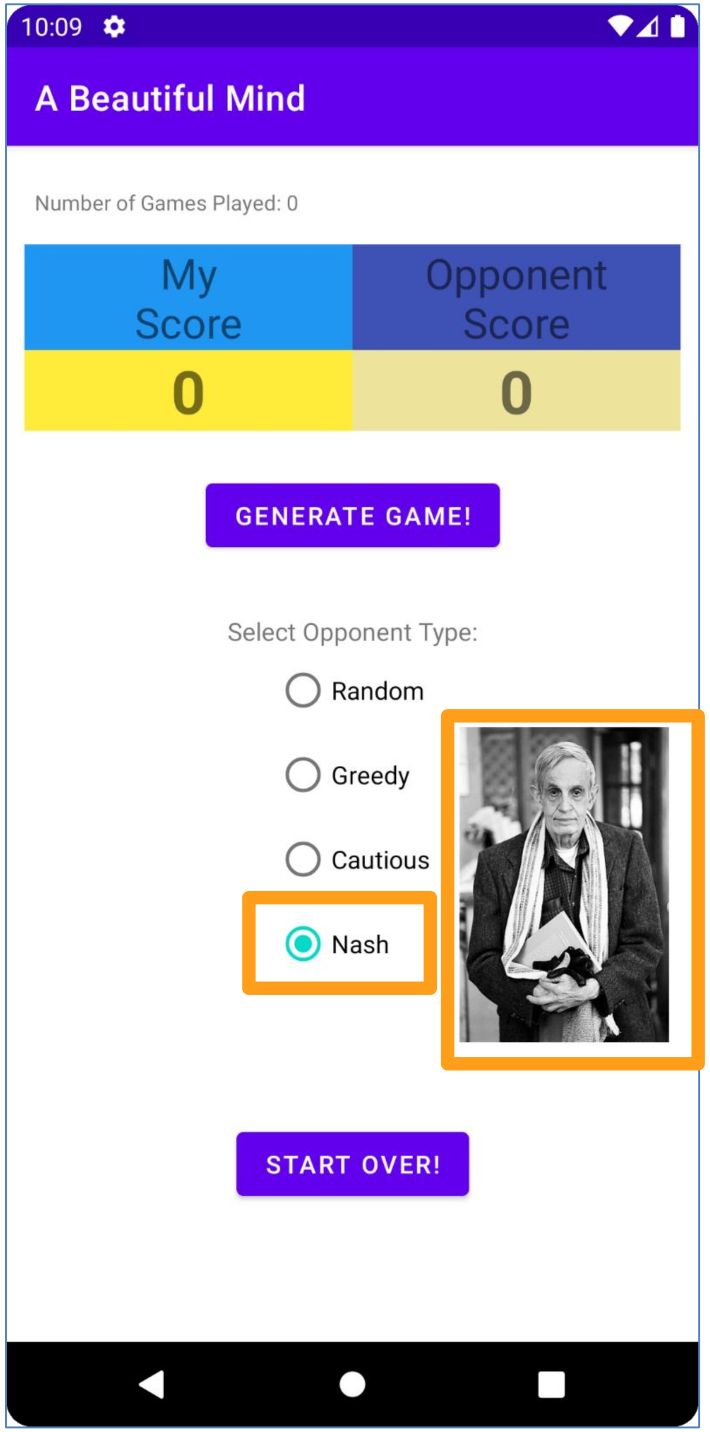
Select Opponent Type:

- Random
- Greedy
- [✓ Cautious]
- Nash

START OVER!

Step 3

- As soon as opponent strategy „Nash“ is selected, an image of John Nash is shown.
  - next to the radio button
  - Image disappears as soon as the user selects a strategy different than „Nash“.



10:09

A Beautiful Mind

Number of Games Played: 0

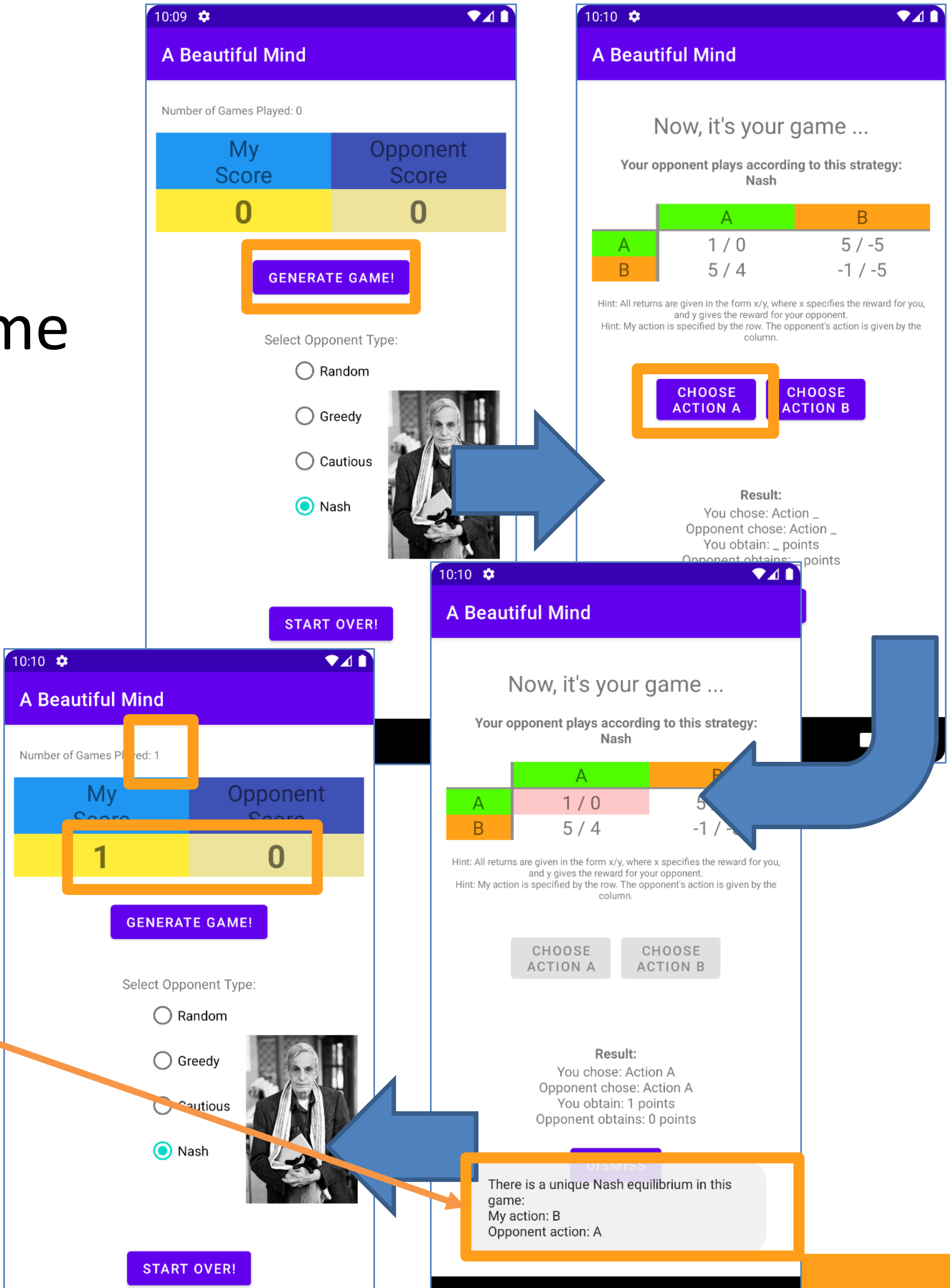
My Score	Opponent Score
0	0

Select Opponent Type:

- ( ) Random
- ( ) Greedy
- ( ) Cautious
- (●) Nash

Step 3

- Example run for a game against a „Nash“ opponent
  - If a single Nash equilibrium exists in the randomly created game, the opponent plays accordingly.
  - A corresponding Toast message shall be displayed as well.



Number of Games Played: 0

My Score	Opponent Score
0	0

Select Opponent Type:
Random
Greedy
Cautious
[✓ Nash]

Now, it's your game ...

Your opponent plays according to this strategy: Nash

	A	B
A	1 / 0	5 / -5
B	5 / 4	-1 / -5

Hint: All returns are given in the form x/y, where x specifies the reward for you, and y gives the reward for your opponent.
Hint: My action is specified by the row. The opponent's action is given by the column.

Result:
You chose: Action _
Opponent chose: Action _
You obtain: _ points
Opponent obtains: _ points

Number of Games Played: 1

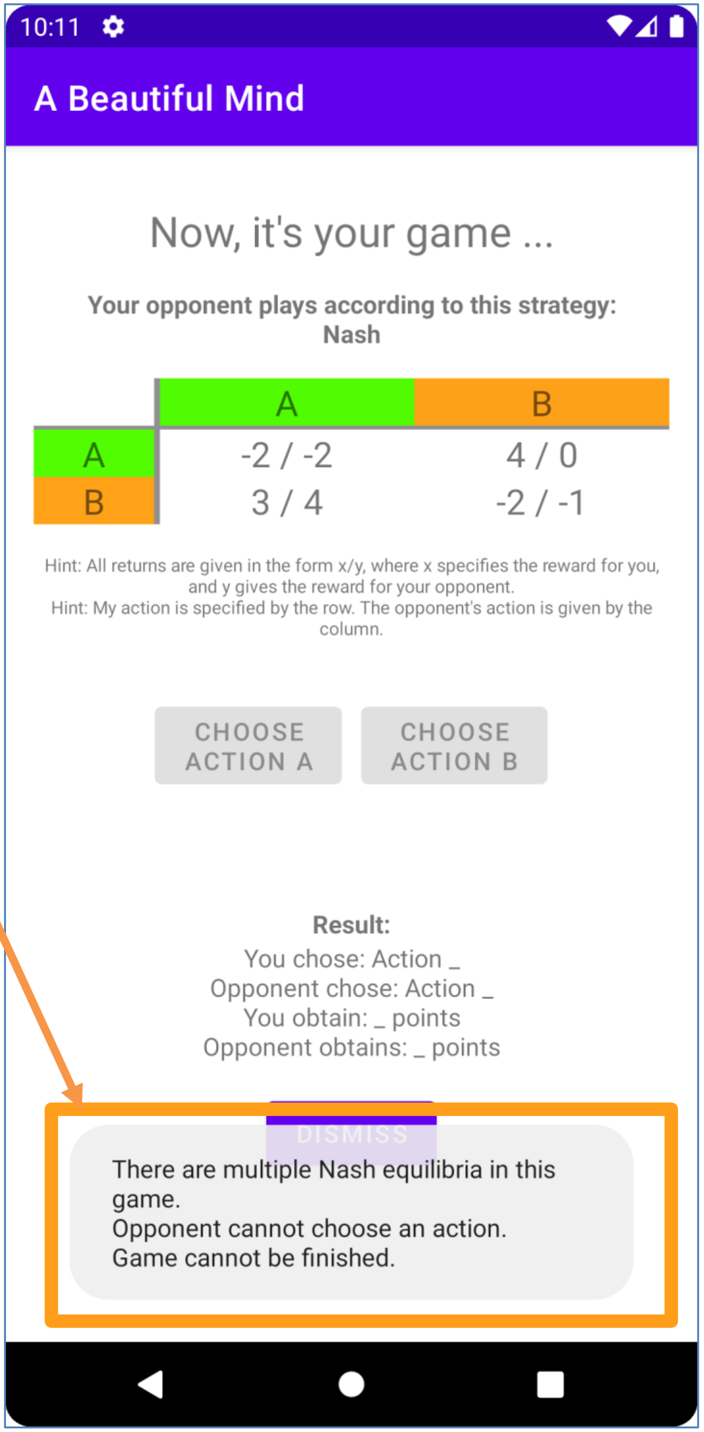
My Score	Opponent Score
1	0

Result:
You chose: Action A
Opponent chose: Action A
You obtain: 1 points
Opponent obtains: 0 points

There is a unique Nash equilibrium in this game:
My action: B
Opponent action: A

Step 3

- There are strategic games in which multiple Nash equlibria exist. In that case
  - a corresponding toast message shall be displayed
  - the „Nash“ opponent does not select an action
  - the games is not counted when clicking „Dismiss“


10:11

A Beautiful Mind

Now, it's your game ...

Your opponent plays according to this strategy:
Nash

	A	B
A	-2 / -2	4 / 0
B	3 / 4	-2 / -1

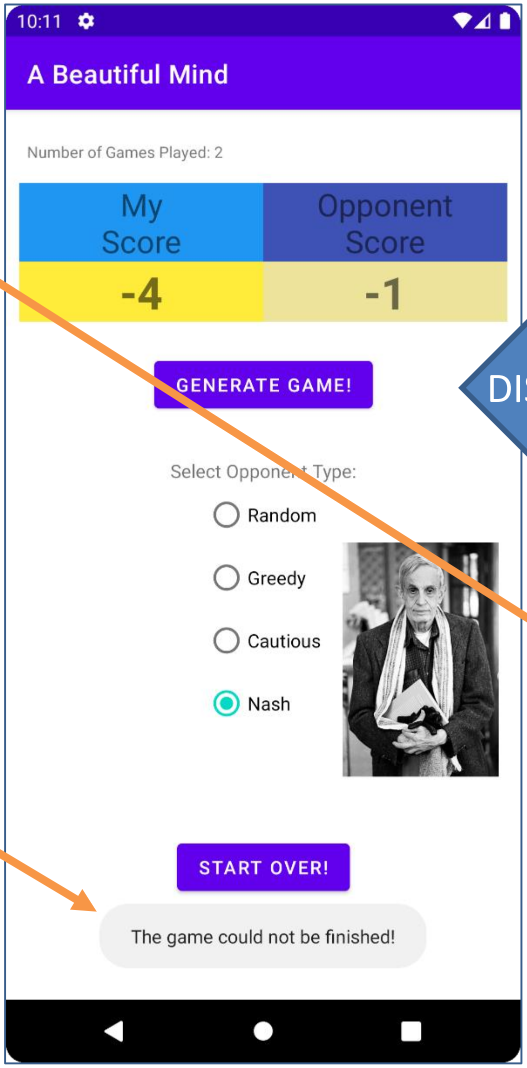
Hint: All returns are given in the form x/y, where x specifies the reward for you, and y gives the reward for your opponent.
Hint: My action is specified by the row. The opponent's action is given by the column.

Result:
You chose: Action _
Opponent chose: Action _
You obtain: _ points
Opponent obtains: _ points

There are multiple Nash equilibria in this game.
Opponent cannot choose an action.
Game cannot be finished.

Step 3

- There are stochastic games in which multiple Nash equlibria exist. In that case
  - a corresponding toast message shall be displayed
  - the „Nash“ opponent does not select an action
  - the games is not counted when clicking „Dismiss“ (scores not changed)
  - a corresponding toast message is displayed after havin returned to the main activity

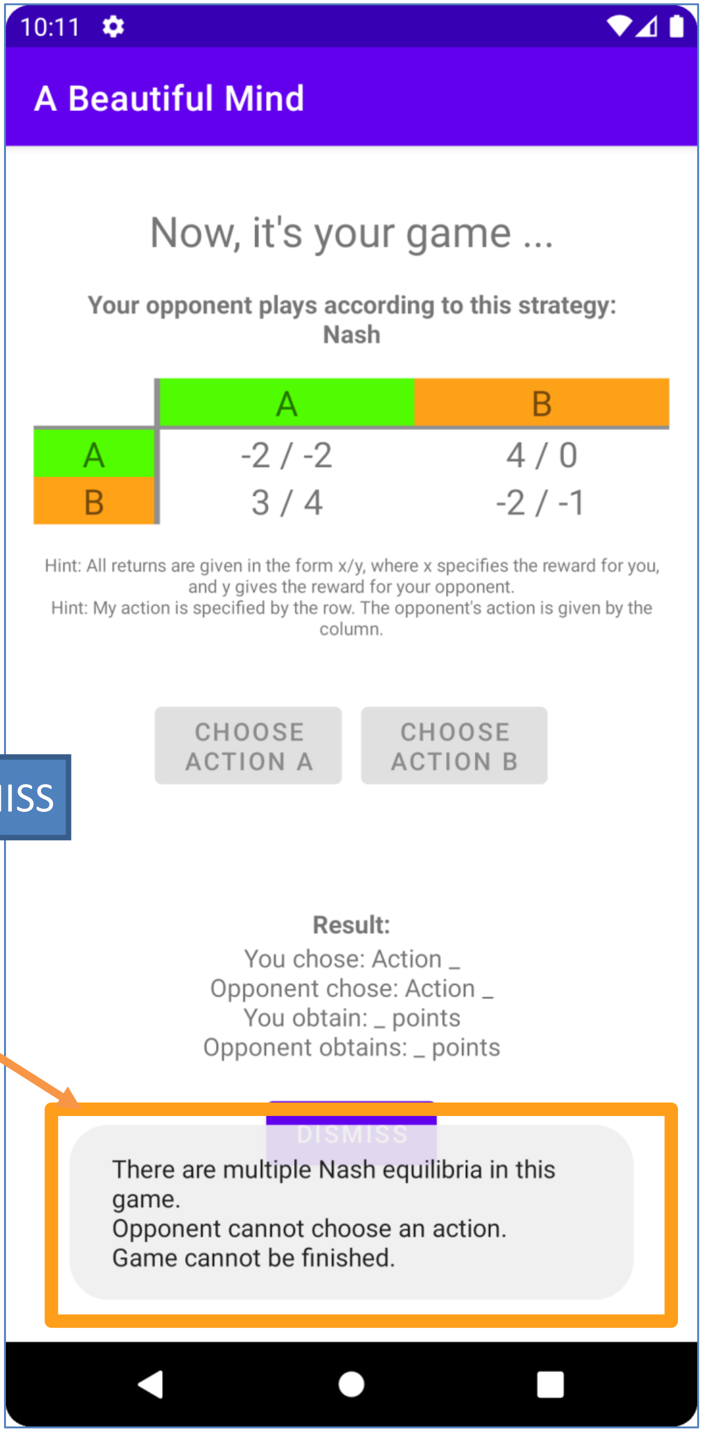

10:11

A Beautiful Mind

Number of Games Played: 2

My Score	Opponent Score
-4	-1

The game could not be finished!


10:11

A Beautiful Mind

Now, it's your game ...

Your opponent plays according to this strategy:
Nash

	A	B
A	-2 / -2	4 / 0
B	3 / 4	-2 / -1

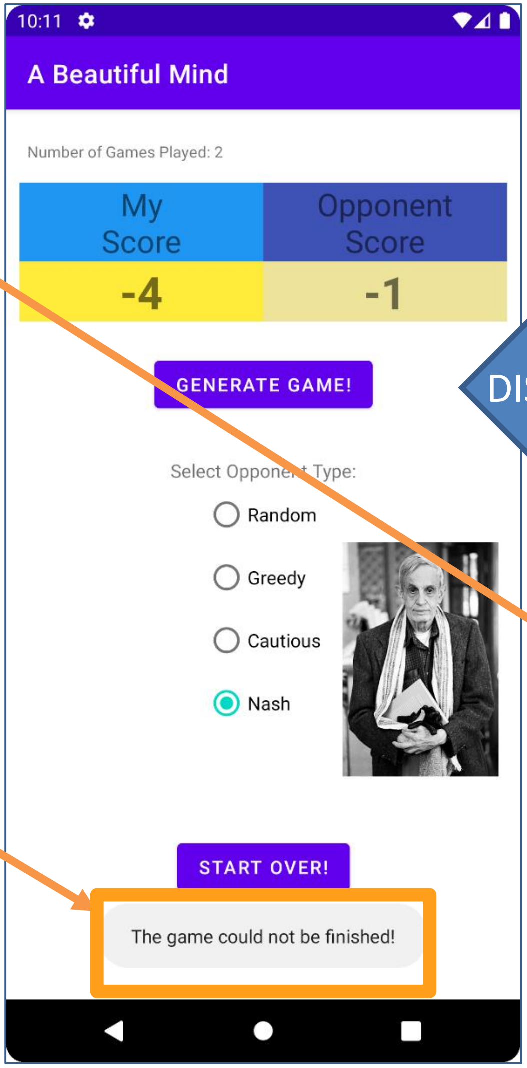
Hint: All returns are given in the form x/y, where x specifies the reward for you, and y gives the reward for your opponent.
Hint: My action is specified by the row. The opponent's action is given by the column.

Result:
You chose: Action _
Opponent chose: Action _
You obtain: _ points
Opponent obtains: _ points

There are multiple Nash equilibria in this game.
Opponent cannot choose an action.
Game cannot be finished.

Step 3

- There are stochastic games in which no Nash equlibria exist. In that case
  - a corresponding toast message shall be displayed
  - the „Nash“ opponent does not select an action
  - the games is not counted when clicking „Dismiss“ (scores not changed)
  - a corresponding toast message is displayed after havin returned to the main activity



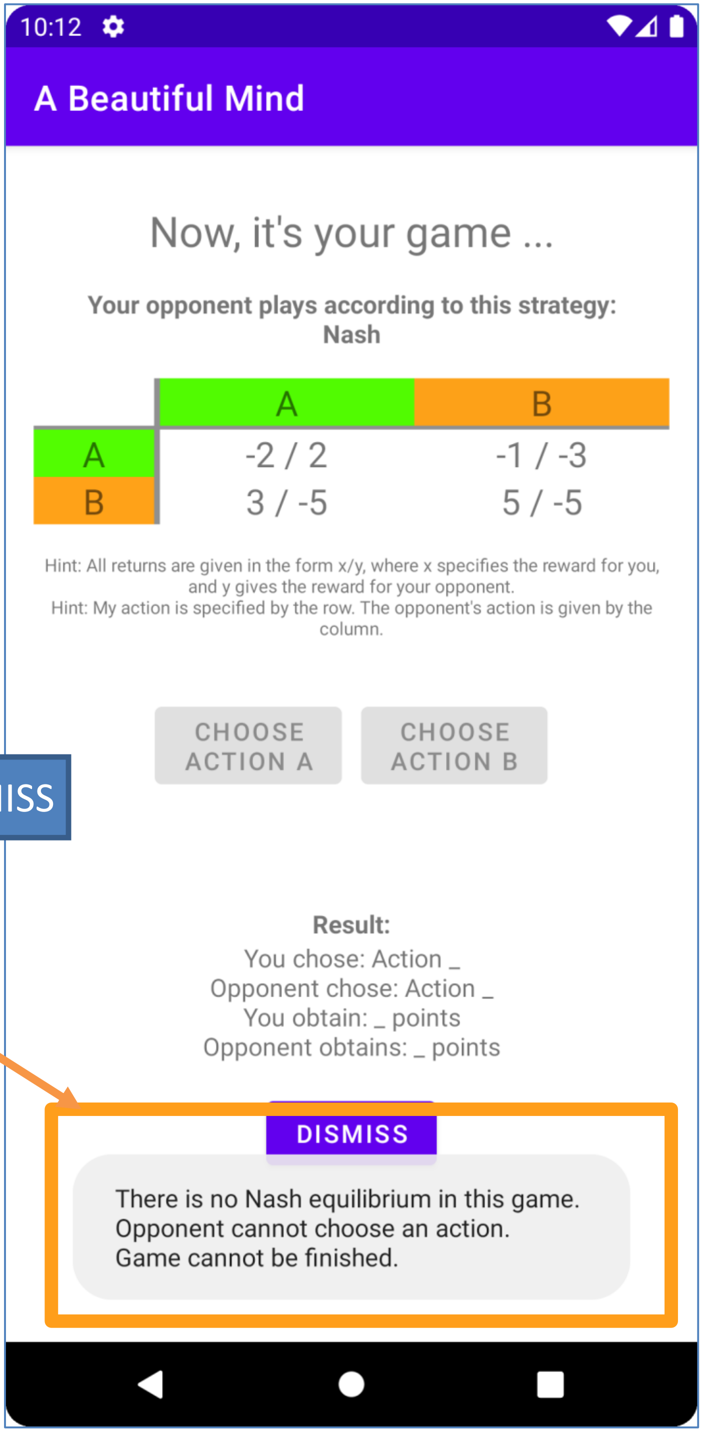
10:11

A Beautiful Mind

Number of Games Played: 2

My Score	Opponent Score
-4	-1

The game could not be finished!



10:12

A Beautiful Mind

Now, it's your game ...

Your opponent plays according to this strategy: Nash

	A	B
A	-2 / 2	-1 / -3
B	3 / -5	5 / -5

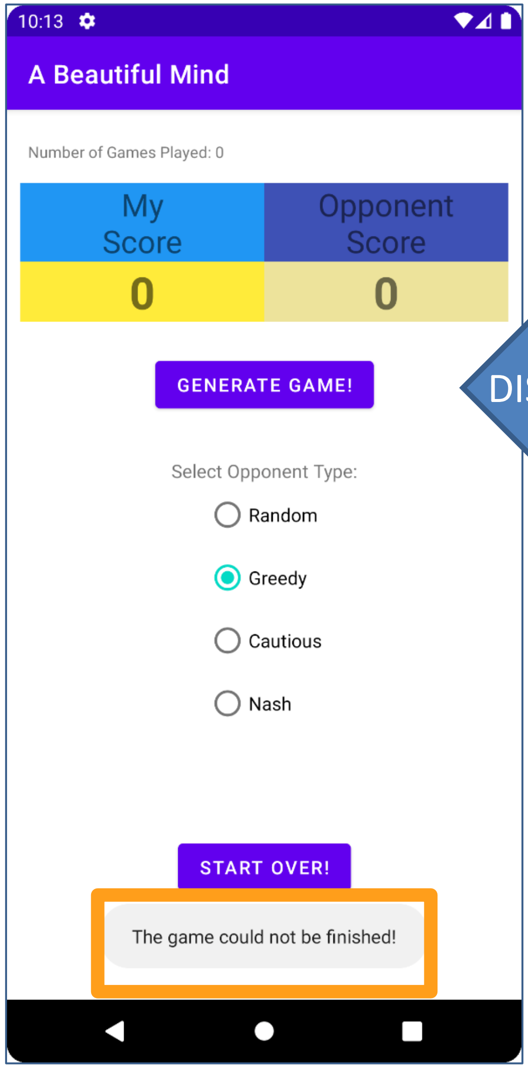
Hint: All returns are given in the form x/y, where x specifies the reward for you, and y gives the reward for your opponent.
Hint: My action is specified by the row. The opponent's action is given by the column.

Result:
You chose: Action _
Opponent chose: Action _
You obtain: _ points
Opponent obtains: _ points

There is no Nash equilibrium in this game.
Opponent cannot choose an action.
Game cannot be finished.

Step 3

- It shall also be possible for the human player to abort any game by clicking „Dismiss“ without having selected an action beforehand.
  - Here is an example where the user dismisses against a greedy opponent.



10:13

A Beautiful Mind

Number of Games Played: 0

My Score	Opponent Score
0	0

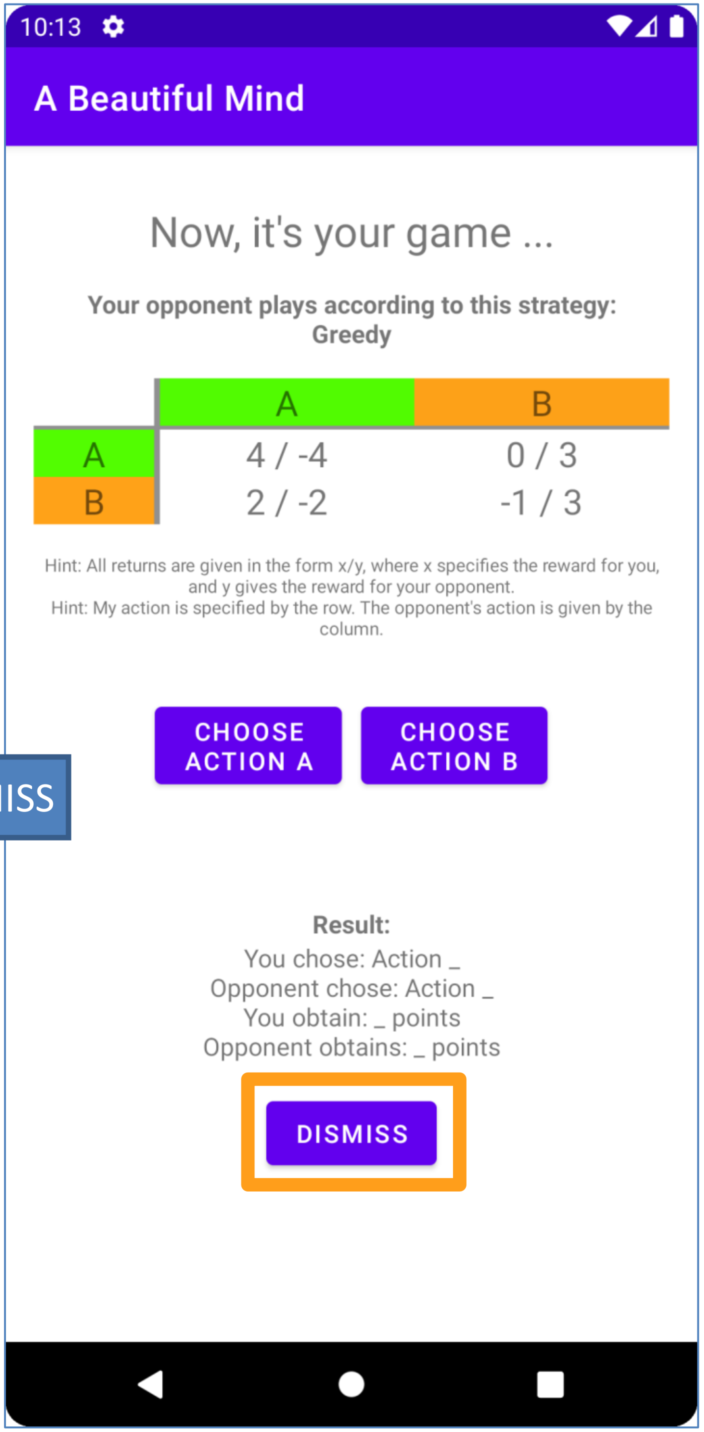
GENERATE GAME!

Select Opponent Type:

- Random
- [✓ Greedy]
- Cautious
- Nash

START OVER!

The game could not be finished!



10:13

A Beautiful Mind

Now, it's your game ...

Your opponent plays according to this strategy: Greedy

	A	B
A	4 / -4	0 / 3
B	2 / -2	-1 / 3

Hint: All returns are given in the form x/y, where x specifies the reward for you, and y gives the reward for your opponent.
Hint: My action is specified by the row. The opponent's action is given by the column.

CHOOSE ACTION A | CHOOSE ACTION B

Result:
You chose: Action _
Opponent chose: Action _
You obtain: _ points
Opponent obtains: _ points

DISMISS

Step 4

- Add functionality to your app according to the description on the next slides.
- All points are optional, but with this additional functionality you can compensate for errors or missing functionality from the parts before.
- Test and debug the app as described before.

Step 4 (Optional Requirements)

- Extend your app such that it properly handles and visualizes screen orientation changes
  - between portrait and landscape mode
  - for both activities
- Take care that all entered values (selected radio buttons, contents of text views) are kept and visualized appropriately, after the rotation has been changed.
- Also, (background) colors shall be kept (e.g. in the game screen).

Step 4 (Optional Requirements)

- Add settings to your app.
- Define the minimum and maximum value of the interval from which utility values are drawn (currently: -5 and 5) and allow the user to specify these values using settings.
  - Take care, that „meaningful“ values are entered.

Step 4
(Optional Requirements)

- Extend the game from two actions (a and b) to a larger set of actions: A={a,b,c,d,…}
  - You may use a fixed number of actions, |A|>2.
  - Or you may make the number of available actions configurable using settings, as well, with a value for |A| from some predefined interval, e.g. from [2,6].
- Take care that the layout of all game elements is still clear and lucid.

Demonstration of Your First App

- The date for the demonstration/attestation will be scheduled using campUAS!
- You have to be available at that date and in time!
- You must be ready to demonstrate your app by
  - explaining the functionality
  - answering question
  - demonstrating the functioning of your app using an emulation (Pixel 3a) or a real device
- You must be able to explain
  - how you realized the functionality
  - typical steps during the development of your app