

## SPOP Puzzle Solver Sprawozdanie

Celem projektu było zaimplementowanie w języku Haskell solvera łamigłówek graficznych opisanych w temacie projektu. Program składa się z dwóch głównych części.

### **Część pierwsza – wczytywanie pliku:**

Jak sama nazwa wskazuje jest to część odpowiadająca za wczytanie łamigłówki w formie tekstowej, ze ścieżki podanej przez użytkownika. Po wczytaniu odbywa się w niej konwersja z typu String na typ `[[Int]]`.

### **Część druga – implementacja algorytmu:**

Został w niej częściowo zaimplementowany algorytm odpowiadający za rozwiązywanie łamigłówek. Opiera się on na dwóch podstawowych podejściach.

- Zakreślaniu części wspólnej otrzymanej poprzez wydobywanie części wspólnej maksymalnie prawego i maksymalnie lewego rozwiązania.
- Drugim podejściem jest, jeżeli już mamy częściowo uzupełnioną łamigłówkę, wykreślenie pól które znajdują się poza zasięgiem (z różnych przyczyn).

W związku z tym w kodzie, główną metodą jest funkcja *solvePuzzle*. Następnie, począwszy od komentarza „--ROWS” jest umieszczony kod odpowiadający za zakreślanie odpowiednich pól na podstawie danych dotyczących wierszy. Kolejna sekcja zaczynająca się od komentarza „--COLUMNS” dotyczy wypełnianie pól na podstawie danych dotyczących kolumn.

Niestety Kolejny etap, nie został zaimplementowany z braku czasu. (Jest on znacznie bardziej skomplikowany i wymaga głębokiego przemyślenia, jeżeli chcielibyśmy wziąć pod uwagę wszystkie przypadki które mogą się w nim pojawić, zwłaszcza jeżeli dopuszczamy pojawianie się w jednym wierszu bądź kolumnie kilku grup zamalowanych kratek.) Na szczęście wykorzystanie zakreślania części wspólnych pozwala na rozwiązywanie prostszych łamigłówek. Pozwala też częściowo wypełnić te bardziej skomplikowane.

### **Zachowanie programu:**

Program zachowuje się zgodnie z oczekiwaniami. W trakcie tworzenia algorytmu wyszło na jaw, że chcąc uniknąć rozwiązania metodą „brutal force” rozwiązanie znacznie się komplikuje.

Do imitowania planszy łamigłówki użyto struktury *Matrix* z modułu *Data.Matrix*.