

## Specyfikacja problemu harmonogramu wersja 2:

### Realizacja funkcjonalności:

Algorytm będzie budowany na podstawie app\_v\_1/code, który realizuje funkcjonalność zawartą w specyfikacji wersja 1. Do tej pory algorytm realizował logikę algorytmu genetycznego z ograniczeniem aby każdy student uczestniczył w trzech zajęciach – uwzględnione to było podczas konstruowania chromosomu. Wersja 2 algorytmu będzie miała narzucone ograniczenia:

- (2) należy zachować ograniczenia liczby miejsc dla danego przedmiotu
- (3) student nie dostał przydziału do zajęć realizowanych o tej samej porze i w tym samym dniu,
- (4) należy uniknąć sytuacji nakładania się przedmiotów obieralnych z przedmiotami podstawowymi – zależącymi od grup podstawowych,
- (5) Niektóre przedmioty tworzą grupę przedmiotów „wspierających się” tzn. wybór jednego pociąga wybór drugiego,

Podczas realizacji algorytmu przyjmujemy, że każdy przedmiot obieralny jest realizowany w jednym terminie.

### Zdefiniowanie struktury danych:

- Dane wejściowe: jedna macierz wymiaru  $n \times m$  ( gdzie  $n$  to liczb studentów, natomiast  $m$  to liczba przedmiotów). Każdy wiersz zawiera pewną permutację liczb od 1... $m$  (wagi dla przedmiotu)
- Dane wejściowe są definiowane losowo w programie. Należy jednak wypisać tą macierz.
- Dane operacyjne: algorytm operuje na macierzy  $n \times m$ , wiersz reprezentujący studenta zawiera wyłącznie wartości 0 (oznacza, że student NIE dostał się na dany przedmiot) lub 1(oznacza, że student dostał się na dany przedmiot). Suma wartości w danym wierszu jest równa 3 (każdy student może dostać się TYLKO na 3 przedmioty)
- Przetwarzanie danych operacyjnych: Dane operacyjne mogą brać udział w procesie krzyżowania i mutacji.
  - a) Krzyżowanie – będzie polegało na użyciu maski czyli wektora binarnego o długości  $n$ . Jeżeli pierwsza wartość wektora jest równa 1 to pobieramy pierwszy wiersz od pierwszego rodzica, natomiast jeśli równa 0 to pobieramy pierwszy wiersz od drugiego rodzica i wstawiamy do potomka. Dla kolejnych wartości współrzędnych wektora bierzemy pod uwagę kolejne wiersze. Podczas jednej operacji krzyżowania otrzymujemy jednego potomka.
  - b) Mutacja polega na wylosowaniu numeru wiersza i wstawieniu w miejscu jego losowo wygenerowanych danych spełniających założenia Dane operacyjne.
- Funkcja celu jest obliczana jako iloczyn Hadamarda macierzy wejściowej i macierzy operacyjnej
- Ograniczenia:
  - każdy student może uczęszczać na 3 przedmioty – uwzględnione podczas krzyżowania i mutacji
  - (2) ograniczenia liczby miejsc na dany przedmiot będą z góry określone przez dane wejściowe, a macierze nie spełniające tych ograniczeń będą odrzucane. Np. iterując po wierszach dla zadanej kolumny np.2 liczba studentów ma być poniżej lub równa 50
  - (3) niektóre przedmioty obieralne niestety się nakładają – czyli wzajemnie się wykluczają – w celu weryfikacji zgodności zastosujemy warunek sprawdzający czy w danym wierszu nie został wygenerowany wektor, który definiuje przynależność do tych dwóch przedmiotów.

Ograniczymy się tylko do 2 przedmiotów które nie mogą zostać przypisane do jednego ucznia

-(4) każdy student będzie miał przypisany indeks grupy podstawowej. Przynależności do grupy podstawowej będzie definiował wektor o wymiarze od 1...n Zadanie będzie polegało na zastosowaniu maski, czyli wektora od 1...m który będzie odpowiedzialny za weryfikację zgodności z planem podstawowym przedmiotu wybranego. Masek jest tyle ile grup podstawowych, czyli  $n/15$  i każda jest indywidualna dla każdej grupy

-przydziału będziemy dokonywali

-(5) niektóre przedmioty obieralne muszą współistnieć razem – czyli wzajemnie się wspierają – w celu weryfikacji zgodności zastosujemy warunek sprawdzający czy w danym wierszu został wygenerowany wektor, który definiuje przynależność do tych dwóch przedmiotów.

Ograniczymy się tylko do 2 przedmiotów które muszą zostać przypisane do jednego ucznia

- Warunek stopu: Liczba pokoleń np. 100, 300
- Parametry sterujące algorytmem:
  - liczność pokolenia – wartość stałą
  - procent pokoleń należących do elity
  - szansa mutacji
  - procent chromosomów ulegającym mutacji
- Dane wyjściowe: Numer Iteracji, Macierz wyjściowa  $n \times m$  - będąca najlepszym wyborem – uzyskana ze zbioru danych operacyjnych, wartość funkcji celu.