

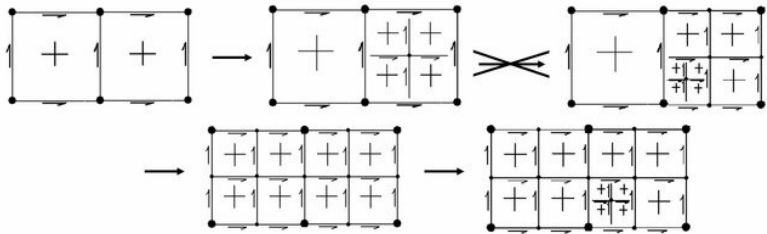
Hypergraph grammar model for the mesh
adaptation for high-order polygonal discontinuous
Petrov-Galerkin (PolyDPG) method

November 16, 2025

Introduction

- Finite Element Method allows for approximation of solutions of some Partial Differential Equations
- The finite element mesh consists of finite elements along with shape functions corresponding to the nodes, edges, and interiors of these elements.
- The approximation of the solution is a linear combination of these shape functions. To enhance solution accuracy, some finite elements may be subdivided into smaller elements.

Hypergraph grammar for mesh refinements for quadrilateral, pentagonal and hexagonal elements for FEM



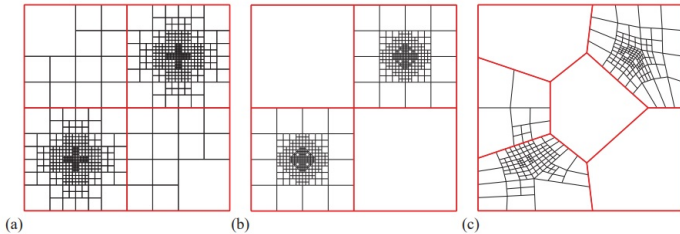
1-irregularity rule An element can be broken only once without breaking its neighbors. (ON ONE EDGE CAN BE AT MOST ONE HANGING NODE)

High-order polygonal discontinuous Petrov-Galerkin (PolyDPG) method - mesh adaptations

High-order polygonal discontinuous Petrov-Galerkin (PolyDPG) method - mesh adaptations

- During the refinement of the traditional quadrilateral elements (like in FEM), typically hanging nodes occur in the mesh.
- Edges cannot have more than one hanging node - an element can be broken only once without breaking neighboring big elements (1-irregularity rule)
- An important advantage of the polygonal elements used in PolyDPG is that they naturally embrace hanging nodes, because they merely represent that a polygon has an extra edge collinear with another edge

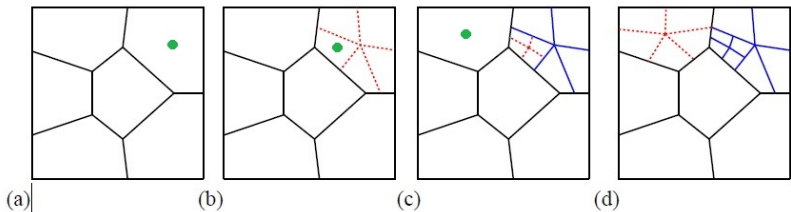
High-order polygonal discontinuous Petrov-Galerkin (PolyDPG) method - mesh adaptations



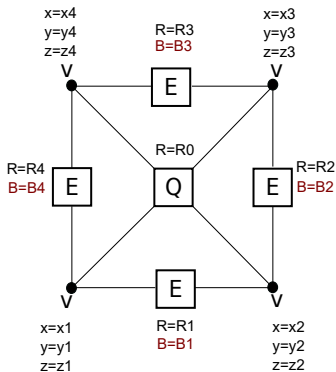
Three h -adaptively refined meshes (the red line represents the initial mesh) for the manufactured solution
(a) traditional quadtree meshes via constrained nodes; (b) quadrilateral mesh using the polygonal adaptive strategy; (c) polygonal mesh using the polygonal adaptive strategy.

High-order polygonal discontinuous Petrov-Galerkin (PolyDPG) method - mesh adaptations

- (a) the initial mesh with element denoted to refinement
- (b) marked element is broken into quadrilaterals by using the centroid and edge midpoints
- (c) marked element is refined
- (d) if a "big" neighbor element needs to be refined, it is split into quadrilaterals assuming all adjacent collinear edges constitute a single edge (the "combined edge" is used in the calculation of the centroid and its midpoint used to place the new quadrilateral node).

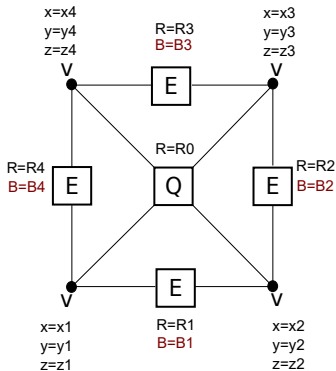


Hypergraph representations of quadrilateral elements for PolyDPG



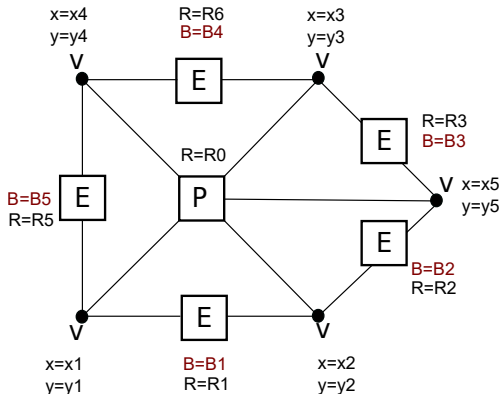
- The hyperedges with label **E** represent edges of the polygon
- attribute **B** - boolean value defining if the edge is a boundary edge ($B = 1$ for a boundary edge, 0 for a shared one).
- attribute **R** boolean value defining if an edge needs to be broken

Hypergraph representations of quadrilateral elements for PolyDPG



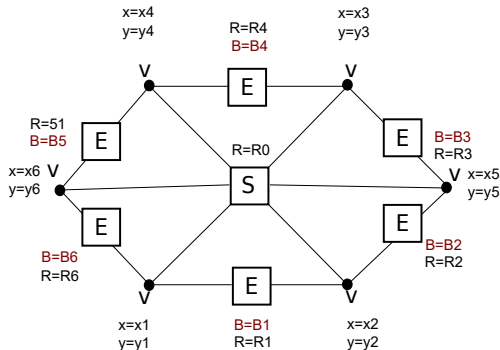
- The hyperedges with label **Q** represent interior nodes of quadrilateral elements
- attribute **R** is a boolean value defining if an element needs to be broken

Hypergraph representations of pentagonal elements for PolyDPG



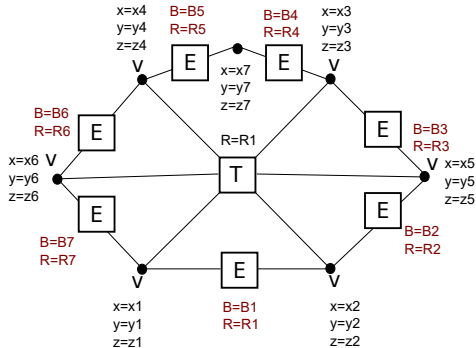
- The hyperedges with label **P** represent interior nodes of pentagonal elements

Hypergraph representations of hexagonal elements for PolyDPG



- The hyperedges with label **S** represent interior nodes of hexagonal elements

Hypergraph representations of septagonal elements for PolyDPG



- The hyperedges with label **T** represent interior nodes of hexagonal elements

Hypergraph grammar for mesh refinements for quadrilateral, pentagonal and hexagonal elements for PolyDPG

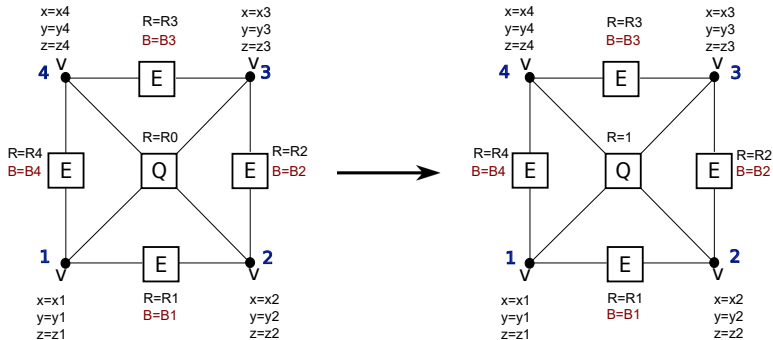
The process of refinement of polygon mesh can be modeled as the four steps process:

- mark element for breaking
- for marked element mark its edges for breaking
- perform breaking of the marked edges
- break the element marked for refinement, with all edges broken

Hypergraph grammar for mesh refinements for quadrilateral, pentagonal and hexagonal elements for PolyDPG

- Production **P0** marks an element for refinement
- it sets value of attribute **R** of the hyperedge with label **Q** to 1

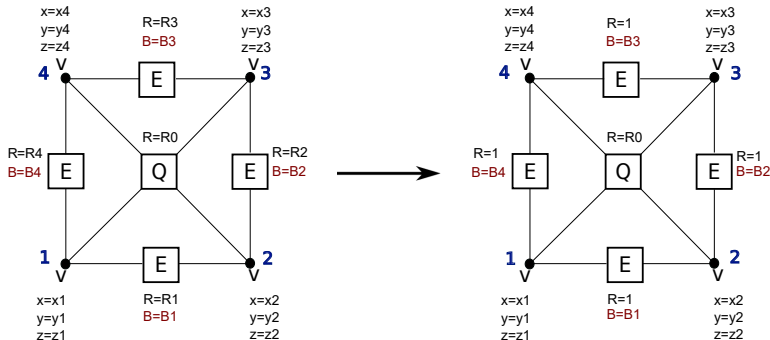
(**R0=0**) and (**RFC**)



Hypergraph grammar for mesh refinements for quadrilateral, pentagonal and hexagonal elements for PolyDPG

- Production **P1** marks edges of **quadrilateral element**, marked for refinement, for breaking
- it sets value of attribute **R** of each hyperedge with label **E** to **1**

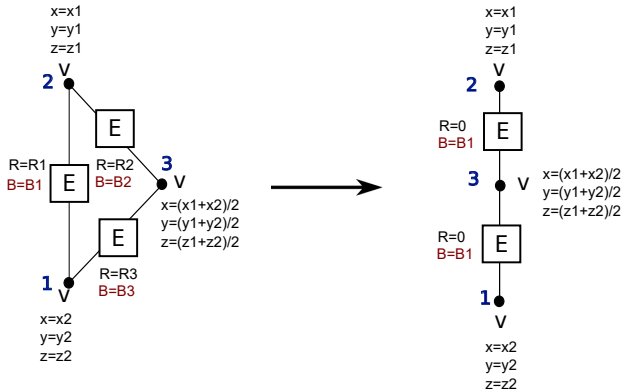
(R0=1)



Hypergraph grammar for mesh refinements for quadrilateral, pentagonal and hexagonal elements for PolyDPG

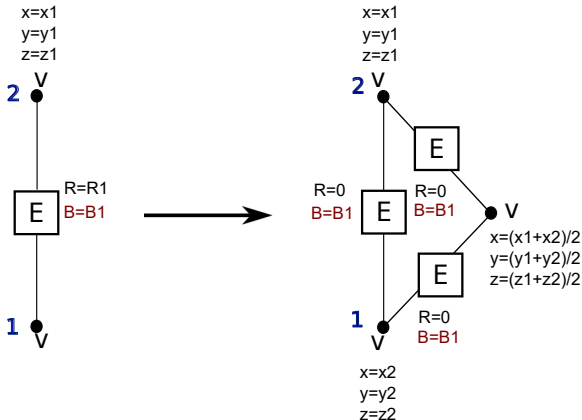
- Production **P2** breaks shared edges marked for refinement, if the edge was already broken by neighboring element
- it sets value of attribute **R** of each hyperedge with label **E** to **0**

(**R1=1**) and (**B1=0**)



Hypergraph grammar for mesh refinements for quadrilateral, pentagonal and hexagonal elements for PolyDPG

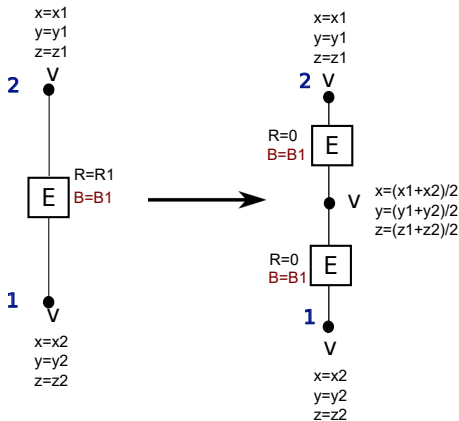
- Production **P3** breaks shared edges marked for refinement, if the edge was not broken by neighboring element
- it sets value of attribute **R** of each hyperedge with label **E** to **0**
(**R1=1**) and (**B1=0**)



Hypergraph grammar for mesh refinements for quadrilateral, pentagonal and hexagonal elements for PolyDPG

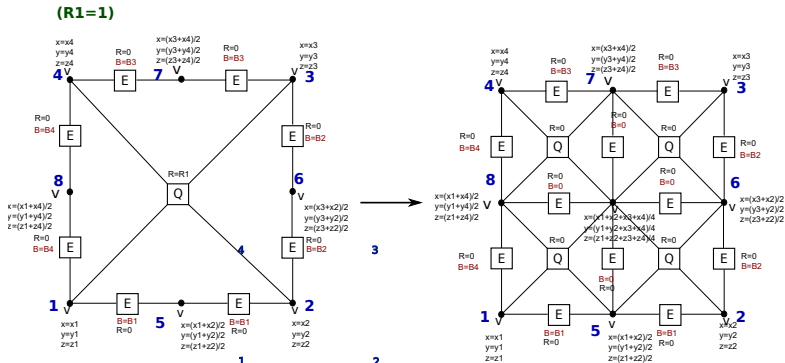
- Production **P4** breaks boundary edges marked for refinement
- it sets value of attribute **R** of each hyperedge with label **E** to 0

(R1=1) and (B1=1)



Hypergraph grammar for mesh refinements for quadrilateral, pentagonal and hexagonal elements for PolyDPG

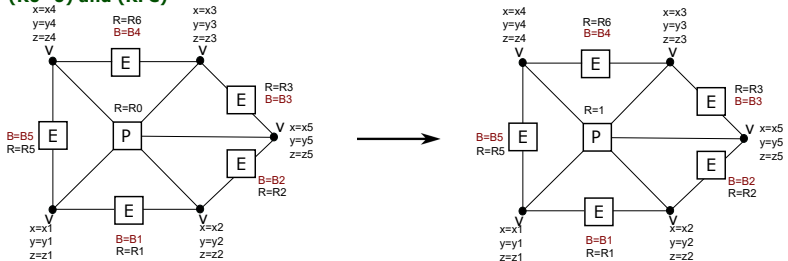
- Production **P5** breaks the quadrilateral element marked for refinement, if all its edges are broken
- it sets value of attribute **R** of new hyperedges with label **Q** to 0



Hypergraph grammar for mesh refinements for quadrilateral, pentagonal and hexagonal elements for PolyDPG

- Production **P6** marks an element for refinement
- it sets value of attribute **R** of the hyperedge with label **P** to 1

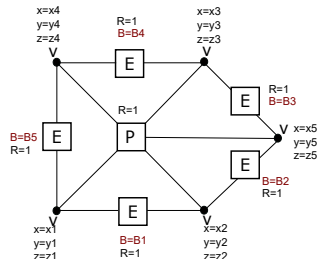
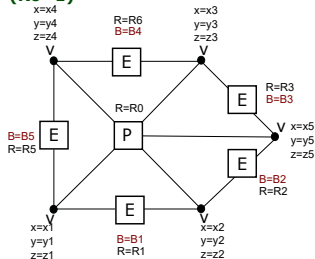
(**R0=0**) and (RFC)



Hypergraph grammar for mesh refinements for quadrilateral, pentagonal and hexagonal elements for PolyDPG

- Production **P7** marks edges of **pentagonal element**, marked for refinement, for breaking
- it sets value of attribute **R** of each hyperedge with label **E** to 1

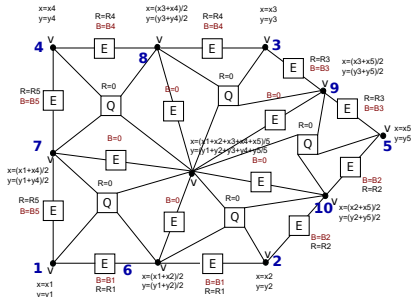
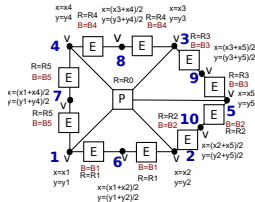
(R0=1)



Hypergraph grammar for mesh refinements for quadrilateral, pentagonal and hexagonal elements for PolyDPG

- Production **P8** breaks the pentagonal element marked for refinement, if all its edges are broken
- it sets value of attribute **R** of new hyperedges with label **Q** to 0

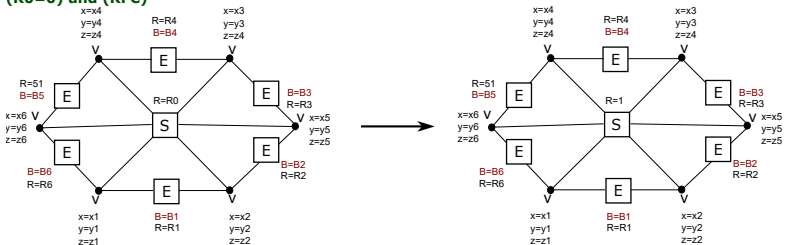
(R0=1)



Hypergraph grammar for mesh refinements for quadrilateral, pentagonal and hexagonal elements for PolyDPG

- Production **P9** marks an element for refinement
- it sets value of attribute **R** of the hyperedge with label **Q** to 1

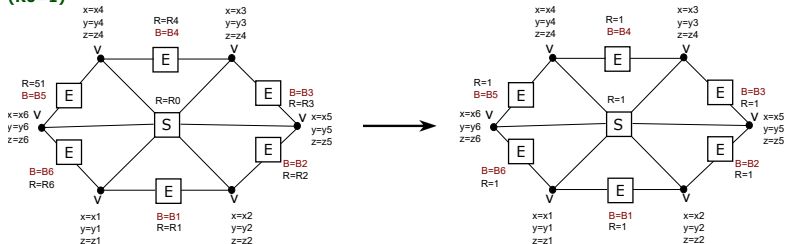
(R0=0) and (RFC)



Hypergraph grammar for mesh refinements for quadrilateral, pentagonal and hexagonal elements for PolyDPG

- Production **P10** marks edges of **hexagonal element**, marked for refinement, for breaking
- it sets value of attribute **R** of each hyperedge with label **E** to 1

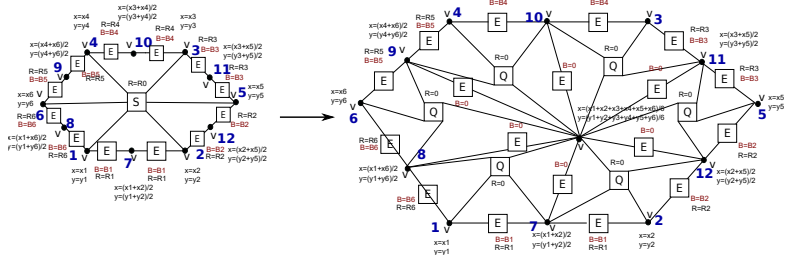
(R0=1)



Hypergraph grammar for mesh refinements for quadrilateral, pentagonal and hexagonal elements for PolyDPG

- Production **P11** breaks the hexagonal element marked for refinement, if all its edges are broken
- it sets value of attribute **R** of new hyperedges with label **Q** to 0

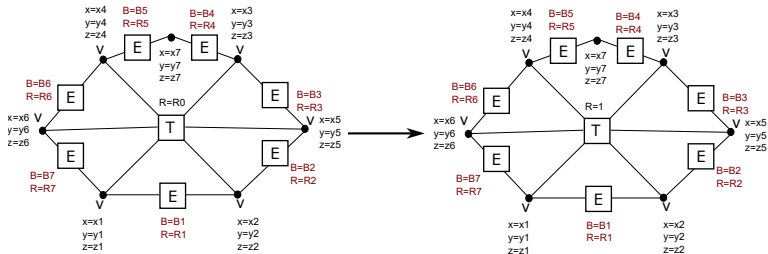
(R0=1)



Hypergraph grammar for mesh refinements for quadrilateral, pentagonal and hexagonal elements for PolyDPG

- Production **P12** marks an element for refinement
- it sets value of attribute **R** of the hyperedge with label **Q** to **1**

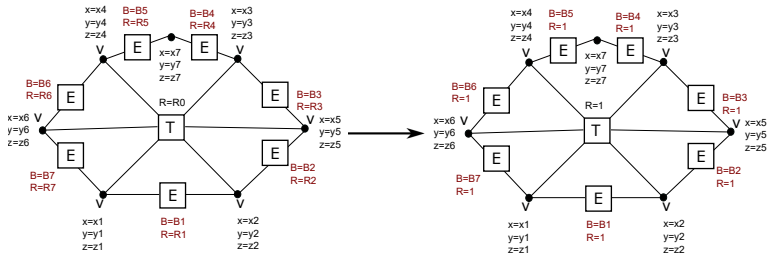
(R0=0) and (RFC)



Hypergraph grammar for mesh refinements for quadrilateral, pentagonal and hexagonal elements for PolyDPG

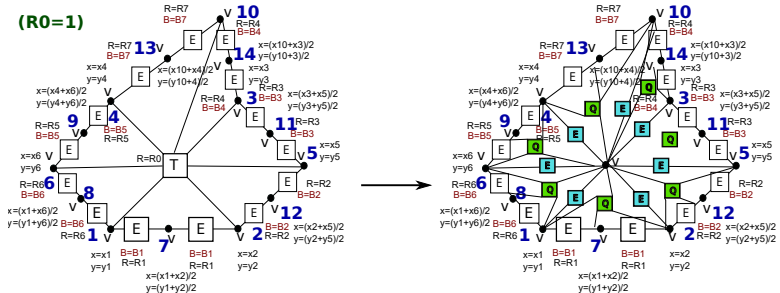
- Production **P13** marks edges of **septagonal element**, marked for refinement, for breaking
- it sets value of attribute **R** of each hyperedge with label **E** to 1

(R0=1)

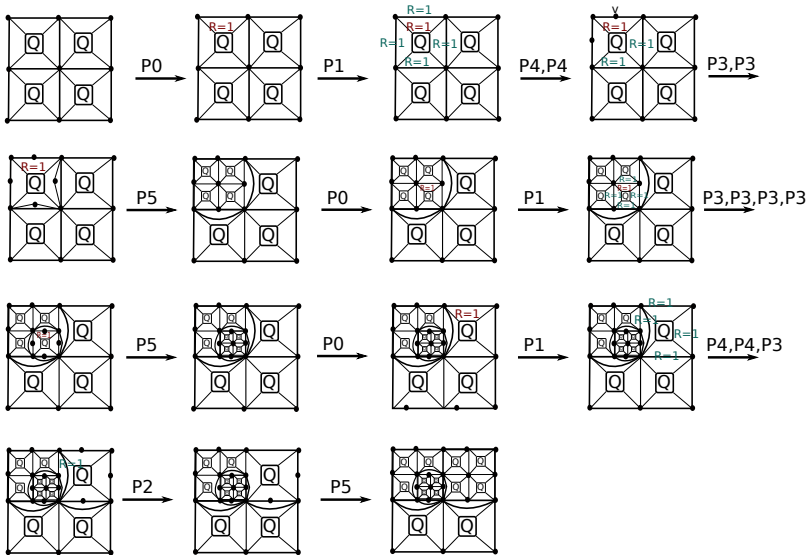


Hypergraph grammar for mesh refinements for quadrilateral, pentagonal and hexagonal elements for PolyDPG

- Production **P14** breaks the septagonal element marked for refinement, if all its edges are broken
- it sets value of attribute **R** of new hyperedges with label **Q** to 0



Hypergraph grammar for mesh refinements for quadrilateral, pentagonal and hexagonal elements for PolyDPG



Zadanie - część pierwsza

- 1 Proszę założyć projekt na githubie i udostępnić go studentom z grupy. Proszę napisać szablon bazowy (klasę bazową) do implementacji wszystkich produkcji . Proszę zaimplementować produkcję (P0) oraz napisać do niej testy jednostkowe wraz z wizualizacją grafu oraz umieścić produkcję oraz testy w githubie.
- 2 Proszę zaimplementować produkcje (P1) i (P5) oraz napisać do nich testy jednostkowe wraz z wizualizacją grafu oraz umieścić produkcję oraz testy w githubie
- 3 Proszę zaimplementować produkcje (P2) i (P6) oraz napisać do nich testy jednostkowe wraz z wizualizacją grafu oraz umieścić produkcję oraz testy w githubie
- 4 Proszę zaimplementować produkcje (P3) i (P7) oraz napisać do nich testy jednostkowe wraz z wizualizacją grafu oraz umieścić produkcję oraz testy w githubie

Zadanie - część pierwsza

- 5 Proszę zaimplementować produkcje (P4) i (P8) oraz napisać do nich testy jednostkowe wraz z wizualizacją grafu oraz umieścić produkcję oraz testy w githubie
- 6 Proszę zaimplementować produkcje (P9) i (P10) oraz napisać do nich testy jednostkowe wraz z wizualizacją grafu oraz umieścić produkcję oraz testy w githubie
- 7 Proszę zaimplementować produkcje (P11) i (P12) oraz napisać do nich testy jednostkowe wraz z wizualizacją grafu oraz umieścić produkcję oraz testy w githubie
- 8 Proszę zaimplementować produkcje (P13) i (P14) oraz napisać do nich testy jednostkowe wraz z wizualizacją grafu oraz umieścić produkcję oraz testy w githubie

- ❶ W jakim piszemy języku?
- ❷ W jakiej bibliotece wizualizujemy grafy?
- ❸ Kto (dwie osoby) weźmie Punkt 1 (github i koordynacje)
[proszę zebrać emaile innych studentów]
- ❹ Po dwie osoby na każde zadanie (zapisy)
- ❺ Pytania?

- ❶ Czy produkcja dobrze sprawdza czy graf (podgraf grafu) do którego chcemy zastosować produkcję jest izomorficzny z grafem lewej strony produkcji (czy da się ją wykonać)?
 - ❶ czy da się wykonać produkcję do grafu izomorficznego z grafem lewej strony produkcji
 - ❷ czy zmiana grafu do którego stosujemy produkcję poprzez usunięcie losowego wierzchołka nie psuje tego mechanizmu
 - ❸ czy zmiana grafu do którego stosujemy produkcję poprzez usunięcie losowej krawędzi nie psuje tego mechanizmu
 - ❹ czy zmiana grafu do którego stosujemy produkcję poprzez zmianę etykiety losowego wierzchołka nie psuje tego mechanizmu
 - ❺ czy umieszczenie grafu izomorficznego z grafem lewej strony jako podgrafu większego grafu nie psuje tego mechanizmu

- ② Czy produkcja dobrze się wykonąła?
 - ① czy jeśli graf izomorficzny z grafem lewej strony jest umieszczony jako podgraf większego grafu, to czy produkcja nie „uszkadza” większego grafu
 - ② czy jeśli graf izomorficzny z grafem lewej strony jest umieszczony w jako podgraf większego grafu, to czy produkcja dobrze transformuje osadzenie
 - ③ czy graf izomorficzny z grafem prawej strony jest poprawny (czy ma wszystkie wierzchołki, krawędzie i poprawne etykiety)
 - ④ czy współrzędne nowych wierzchołków w tym grafie są poprawne

- ③ Czy graf po zastosowaniu produkcji dobrze się rysuje?
 - ① czy są wszystkie wierzchołki i krawędzie
 - ② czy wierzchołki są narysowane w poprawnych współrzędnych
 - ③ czy są narysowane etykiety wierzchołków

- ④ Czy zostały przygotowanie różne grafy do testowania
 - ① czy został przygotowany graf izomorficzny z grafem lewej strony produkcji,
 - ② czy został przygotowany graf zawierający graf izomorficzny z grafem lewej strony produkcji
 - ③ czy został przygotowany graf izomorficzny z grafem lewej strony produkcji, który jest niepoprawny (bez jakiegoś wierzchołka)
 - ④ czy został przygotowany graf izomorficzny z grafem lewej strony produkcji, który jest niepoprawny (bez jakiejś krawędzi)
 - ⑤ czy został przygotowany graf izomorficzny z grafem lewej strony produkcji, który jest niepoprawny (z niepoprawną etykietą)
 - ⑥ czy został przygotowany graf izomorficzny z grafem lewej strony produkcji, który jest niepoprawny (z błędnymi współrzędnymi wierzchołków)

- ⑤ Czy wynik uzyskany po zastosowaniu produkcji został dobrze sprawdzony
 - ① czy zostało sprawdzone czy produkcja wykonała się na poprawnym grafie i nie została wykonana na niepoprawnym grafie?
 - ② czy zostało sprawdzone czy jeśli graf lewej strony jest umieszczony w jako podgraf większego grafu, to czy produkcja nie uszkadza większego grafu
 - ③ czy zostało sprawdzone czy jeśli graf lewej strony jest umieszczony jako podgraf większego grafu, to czy produkcja dobrze transformuje osadzenie
 - ④ czy zostało sprawdzone czy graf prawej strony jest poprawny (czy ma wszystkie wierzchołki, krawędzie i poprawne etykiety)
 - ⑤ czy zostało sprawdzone czy współrzędne nowych wierzchołków są poprawne