вариант	ф. номер	група	поток	курс	специалност
K2.1					
Име:					

Писмен изпит по логическо програмиране
25 януари 2022 год.

Да няма лист, на който е писано по повече от една задача!
За **всеки дефиниран** предикат да се попълни подходящият/те шаблон(и):

1. При параметри ..., предикатът ... разпознава дали ...
2. При параметри ..., предикатът ... генерира ... в ...
3. $p(\dots)$ е истина тогава и само тогава, когато ... Следното условие е достатъчно, за да няма зацикляне с предиката: ...

Решения на задачи, в които това отсъства, ще бъдат оценявани с 0 точки.

Зад. 1. Нека $G = (V, E)$ е ориентиран граф. За два върха $v, u \in V$ на G казваме, че *u е съсед на v*, ако $(u, v) \in E$ е ребро на G. Ще наричаме графа G *k-съвършен*, ако за всеки връх $v \in V$ множеството от съседите на v съвпада с множеството от онези върхове $u \in V$, за които има път (незадължително прост) от v до u с дължина точно k. *Представяне* на G наричаме такъв списък $Edges$ от двуелементни списъци, че за всяко ребро $(u, v) \in E$ на G списъкът $[u, v]$ е елемент на $Edges$ и множеството E и списъкът $Edges$ имат един и същ брой елементи.
Да се дефинира на пролог двуместен предикат `pc_Gr`$(Edges, K)$, който по дадени представяне $Edges$ на ориентиран граф G без изолирани върхове и естествено число $K > 1$ разпознава дали G е K-съвършен.

Зад. 2. *Разделяне* на краен списък $L = [a_1, a_2, \dots, a_n]$ наричаме такава двойка (L_1, L_2) от списъци, че $L_1 = [a_{i_1}, a_{i_2}, \dots, a_{i_k}]$, $L_2 = [a_{j_1}, a_{j_2}, \dots, a_{j_{n-k}}]$ и $\{i_1, i_2, \dots, i_k\} \cup \{j_1, j_2, \dots, j_{n-k}\} = \{1, 2, \dots, n\}$. *Разбиване* на естествено число N наричаме списък L от положителни цели числа, чиято сума е N. За едно разбиване L на N казваме, че е *уравновесимо*, ако има такова разделяне (L_1, L_2) на L, че L_1 и L_2 са разбивания на едно и също естествено число.
Да се дефинира на пролог двуместен предикат `equPart`(N, L), който по дадено естествено число N при преудовлетворяване генерира в L последователно всички уравновесими разбивания на N.

Пожелаваме ви приятна и успешна работа!

вариант	ф. номер	група	поток	курс	специалност
K2.2					
Име:					

Писмен изпит по логическо програмиране
25 януари 2022 год.

Да няма лист, на който е писано по повече от една задача!
За **всеки дефиниран** предикат да се попълни подходящият/те шаблон(и):

1. При параметри ..., предикатът ... разпознава дали ...
2. При параметри ..., предикатът ... генерира ... в ...
3. $p(\dots)$ е истина тогава и само тогава, когато ... Следното условие е достатъчно, за да няма зацикляне с предиката: ...

Решения на задачи, в които това отсъства, ще бъдат оценявани с 0 точки.

Зад. 1. Нека $G = (V, E)$ е ориентиран граф. За два върха $v, u \in V$ на G казваме, че *u е съсед на v*, ако $(v, u) \in E$ е ребро на G. Ще наричаме графа G *k-съвършен*, ако за всеки връх $v \in V$ множеството от съседите на v съвпада с множеството от онези върхове $u \in V$, за които има път (незадължително прост) от v до u с дължина точно k. *Представяне* на G наричаме такъв списък $Edges$ от двуелементни списъци, че за всяко ребро $(u, v) \in E$ на G списъкът $[u, v]$ е елемент на $Edges$ и множеството E и списъкът $Edges$ имат един и същ брой елементи.
Да се дефинира на пролог двуместен предикат `pr_Gr`$(Edges, K)$, който по дадени представяне $Edges$ на ориентиран граф G без изолирани върхове и естествено число $K > 1$ разпознава дали G е K-съвършен.

Зад. 2. *Разделяне* на краен списък $L = [a_1, a_2, \dots, a_n]$ наричаме такава двойка (L_1, L_2) от списъци, че $L_1 = [a_{i_1}, a_{i_2}, \dots, a_{i_k}]$, $L_2 = [a_{j_1}, a_{j_2}, \dots, a_{j_{n-k}}]$и $\{i_1, i_2, \dots, i_k\} \cup \{j_1, j_2, \dots, j_{n-k}\} = \{1, 2, \dots, n\}$. *Разбиване* на естествено число N наричаме списък L от положителни цели числа, чиято сума е N. За едно разбиване L на N казваме, че е *балансируемо*, ако има разделяне (L_1, L_2) на L, за което $|N_1 - N_2| = 1$, където L_1 и L_2 са разбивания съответно на N_1 и на N_2.
Да се дефинира на пролог двуместен предикат `balPart`(N, L), който по дадено естествено число N при преудовлетворяване генерира в L последователно всички балансируеми разбивания на N.

Пожелаваме ви приятна и успешна работа!

вариант	ф. номер	група	поток	курс	специалност
K2.1					
Име:					

Писмен изпит по логическо програмиране
25 януари 2022 год.

Да няма лист, на който е писано по повече от една задача!
За **всеки дефиниран** предикат да се попълни подходящият/те шаблон(и):

1. При параметри ..., предикатът ... разпознава дали ...
2. При параметри ..., предикатът ... генерира ... в ...
3. $p(\dots)$ е истина тогава и само тогава, когато ... Следното условие е достатъчно, за да няма зацикляне с предиката: ...

Решения на задачи, в които това отсъства, ще бъдат оценявани с 0 точки.

Зад. 1. Нека $G = (V, E)$ е ориентиран граф. За два върха $v, u \in V$ на G казваме, че *u е съсед на v*, ако $(u, v) \in E$ е ребро на G. Ще наричаме графа G *k-съвършен*, ако за всеки връх $v \in V$ множеството от съседите на v съвпада с множеството от онези върхове $u \in V$, за които има път (незадължително прост) от v до u с дължина точно k. *Представяне* на G наричаме такъв списък $Edges$ от двуелементни списъци, че за всяко ребро $(u, v) \in E$ на G списъкът $[u, v]$ е елемент на $Edges$ и множеството E и списъкът $Edges$ имат един и същ брой елементи.
Да се дефинира на пролог двуместен предикат `pc_Gr`$(Edges, K)$, който по дадени представяне $Edges$ на ориентиран граф G без изолирани върхове и естествено число $K > 1$ разпознава дали G е K-съвършен.

Зад. 2. *Разделяне* на краен списък $L = [a_1, a_2, \dots, a_n]$ наричаме такава двойка (L_1, L_2) от списъци, че $L_1 = [a_{i_1}, a_{i_2}, \dots, a_{i_k}]$, $L_2 = [a_{j_1}, a_{j_2}, \dots, a_{j_{n-k}}]$ и $\{i_1, i_2, \dots, i_k\} \cup \{j_1, j_2, \dots, j_{n-k}\} = \{1, 2, \dots, n\}$. *Разбиване* на естествено число N наричаме списък L от положителни цели числа, чиято сума е N. За едно разбиване L на N казваме, че е *уравновесимо*, ако има такова разделяне (L_1, L_2) на L, че L_1 и L_2 са разбивания на едно и също естествено число.
Да се дефинира на пролог двуместен предикат `equPart`(N, L), който по дадено естествено число N при преудовлетворяване генерира в L последователно всички уравновесими разбивания на N.

Пожелаваме ви приятна и успешна работа!

вариант	ф. номер	група	поток	курс	специалност
K2.2					
Име:					

Писмен изпит по логическо програмиране
25 януари 2022 год.

Да няма лист, на който е писано по повече от една задача!
За **всеки дефиниран** предикат да се попълни подходящият/те шаблон(и):

1. При параметри ..., предикатът ... разпознава дали ...
2. При параметри ..., предикатът ... генерира ... в ...
3. $p(\dots)$ е истина тогава и само тогава, когато ... Следното условие е достатъчно, за да няма зацикляне с предиката: ...

Решения на задачи, в които това отсъства, ще бъдат оценявани с 0 точки.

Зад. 1. Нека $G = (V, E)$ е ориентиран граф. За два върха $v, u \in V$ на G казваме, че *u е съсед на v*, ако $(v, u) \in E$ е ребро на G. Ще наричаме графа G *k-съвършен*, ако за всеки връх $v \in V$ множеството от съседите на v съвпада с множеството от онези върхове $u \in V$, за които има път (незадължително прост) от v до u с дължина точно k. *Представяне* на G наричаме такъв списък $Edges$ от двуелементни списъци, че за всяко ребро $(u, v) \in E$ на G списъкът $[u, v]$ е елемент на $Edges$ и множеството E и списъкът $Edges$ имат един и същ брой елементи.
Да се дефинира на пролог двуместен предикат `pr_Gr`$(Edges, K)$, който по дадени представяне $Edges$ на ориентиран граф G без изолирани върхове и естествено число $K > 1$ разпознава дали G е K-съвършен.

Зад. 2. *Разделяне* на краен списък $L = [a_1, a_2, \dots, a_n]$ наричаме такава двойка (L_1, L_2) от списъци, че $L_1 = [a_{i_1}, a_{i_2}, \dots, a_{i_k}]$, $L_2 = [a_{j_1}, a_{j_2}, \dots, a_{j_{n-k}}]$и $\{i_1, i_2, \dots, i_k\} \cup \{j_1, j_2, \dots, j_{n-k}\} = \{1, 2, \dots, n\}$. *Разбиване* на естествено число N наричаме списък L от положителни цели числа, чиято сума е N. За едно разбиване L на N казваме, че е *балансируемо*, ако има разделяне (L_1, L_2) на L, за което $|N_1 - N_2| = 1$, където L_1 и L_2 са разбивания съответно на N_1 и на N_2.
Да се дефинира на пролог двуместен предикат `balPart`(N, L), който по дадено естествено число N при преудовлетворяване генерира в L последователно всички балансируеми разбивания на N.

Пожелаваме ви приятна и успешна работа!