HO GENT	**Hogeschool Gent – Faculteit Bedrijf en Organisatie FBO** **Toegepaste Informatica 1e jaar – Academiejaar 2019 - 2020** **OEFENINGEN OO Programmeren I – Hoofdstuk 5**

1. Lees een reeks in van 3 gehele getallen: **de operator (via een code), de eerste operand x en de tweede operand y.**

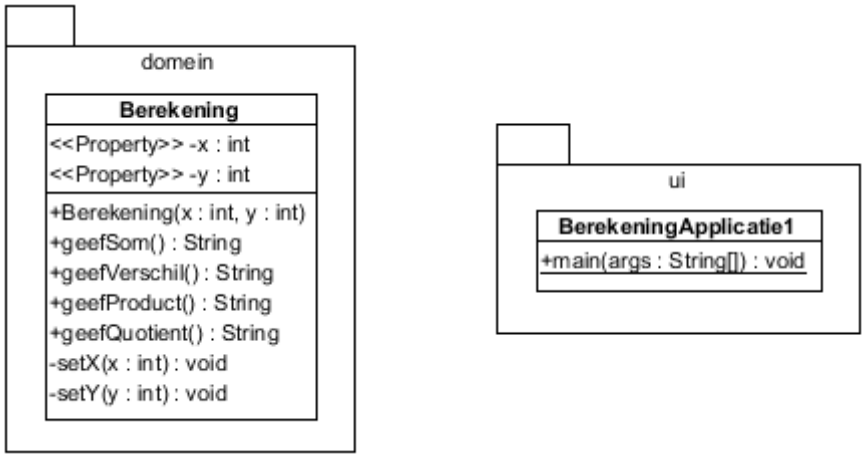
   Volgens de waarde van de code moet er een bepaalde bewerking gebeuren:
   code = 1: x + y
   code = 2: grootste getal - kleinste getal
   code = 3: x * y
   code = 4: grootste getal / kleinste getal (gehele deling!)

   Geef telkens een resultatenstring van de volgende vorm, bijv. bij code = 2:
   Het verschil tussen … en … is ….

   Zorg voor een foutboodschap wanneer de code verschillend is van 1, 2, 3, 4 en –1.
   Als de code gelijk is aan –1, dan sluit de applicatie af.

   Maak gebruik van een **domeinklasse Berekening**
   attributen: x, y
   methoden: constructor met 2 argumenten: x wordt opgevuld met het grootste getal, y met het kleinste getal.
   Methoden geefSom, geefVerschil, geefProduct en geefQuotient. Deze methoden geven telkens een string terug, bv. “Het product van ... en ... is ...”
   Maak ook gebruik van methoden van Math voor het bepalen van het grootste en kleinste getal**.**

   UML:



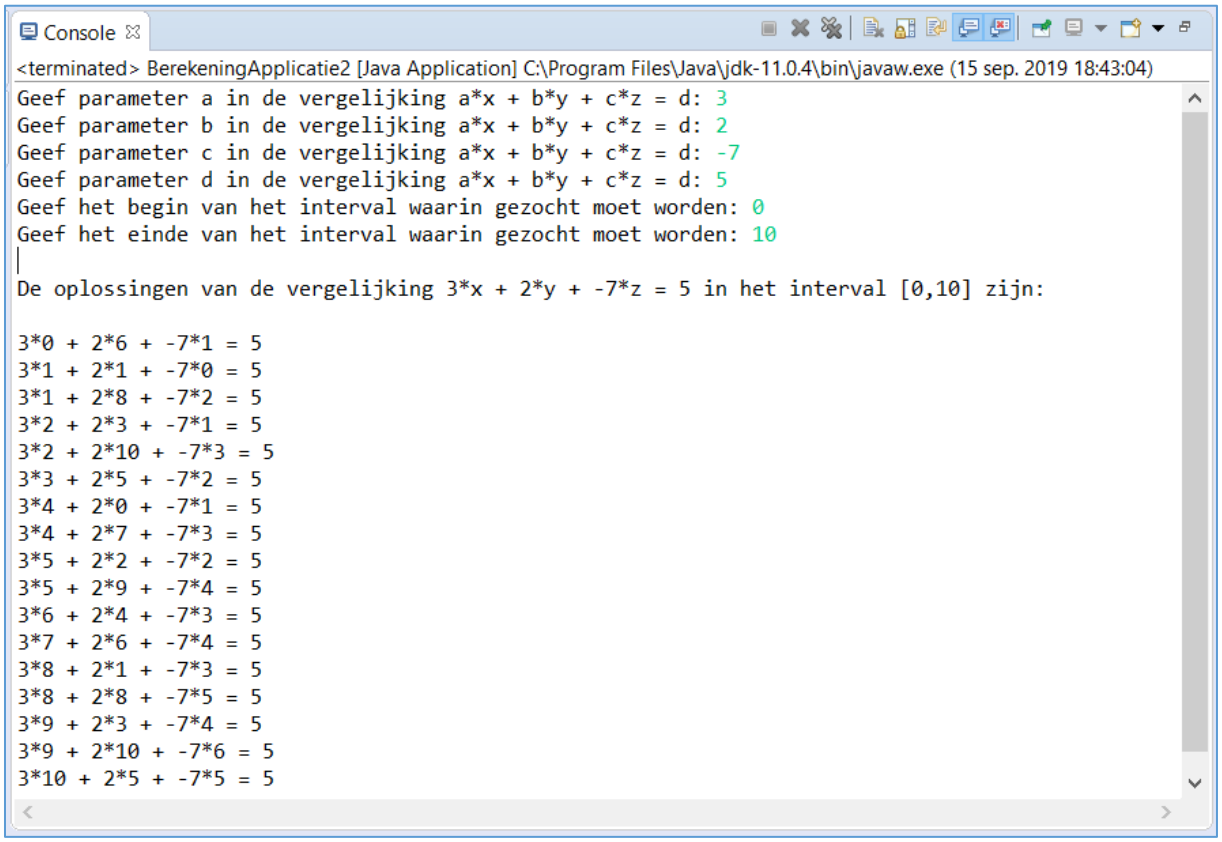
2. Zoek alle oplossingen van de vergelijking

   3x + 2y – 7z = 5 met x, y en z behorend tot [0,10]

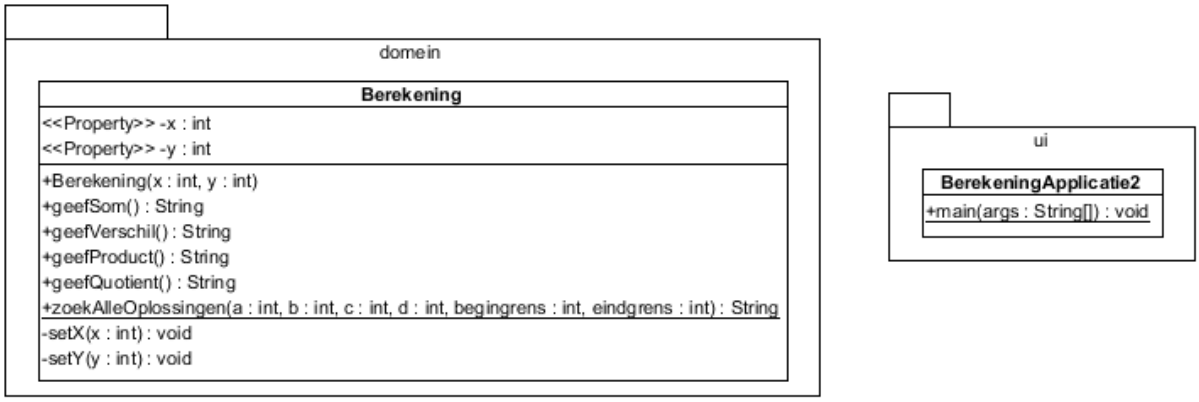
Werk de domeinklasse Berekening verder uit.

Maak de static methode zoekAlleOplossingen(int a, int b, int c, int d, int begingrens, int eindgrens) die een String teruggeeft met alle oplossingen van de vergelijking
ax + by + cz = d met x,y en z in [begingrens,eindgrens]

Uitvoer:

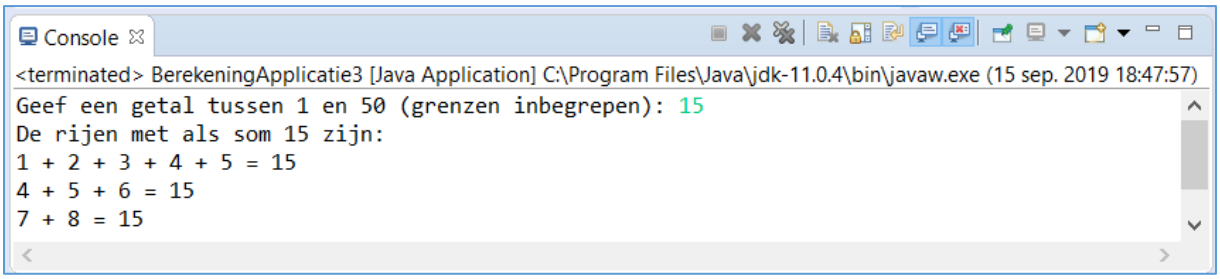


UML:



3. Lees een positief geheel getal x, verschillend van nul en niet groter dan 50, in (+ controle). Bepaal alle rijen van opeenvolgende positieve getallen waarvan de som x is.

Uitvoer:



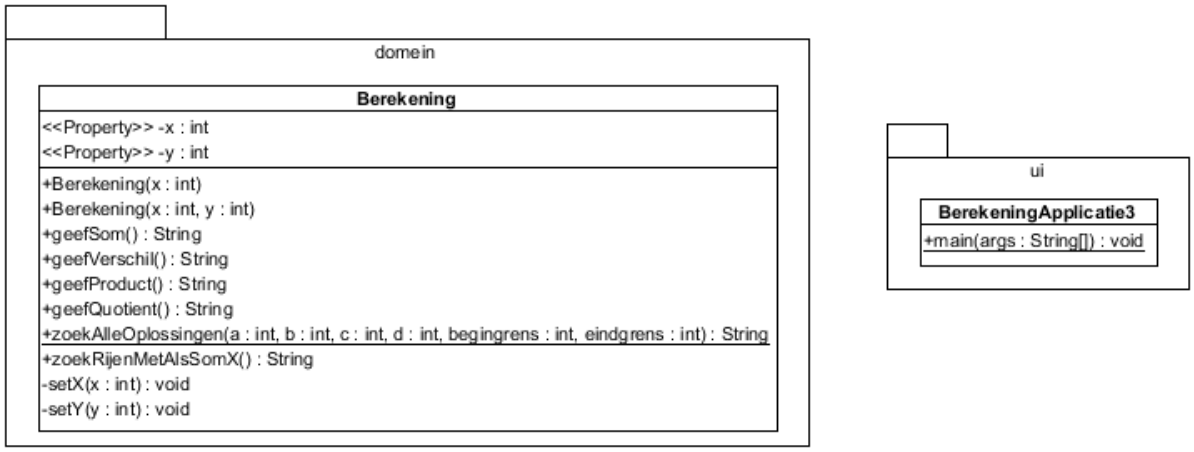
Werk de domeinklasse Berekening verder uit.

Constructor met één argument: x

Maak de methode zoekRijenMetAlsSomX die een String teruggeeft met alle rijen van opeenvolgende positieve getallen waarvan de som x is.

ApplicatieKlasse: de gebruiker moet een getal ingeven tussen 1 en 50, grenzen inbegrepen. Zolang de gebruiker geen geldige waarde ingeeft, wordt er opnieuw een getal gevraagd.

UML:



4. In de **domeinklasse Speler** komen drie attributen, een constructor (met één argument) en vijf methodes. De **attributen** zijn: naam (om de naam van de speler bij te houden), vaknr (een geheel getal dat aanduidt op welk vak de pion van de speler staat) en GROOTSTENR (een constant geheel getal, hier 63, dat het grootste vaknummer op het spelbord aangeeft). Dit laatste attribuut hoort bij de klasse, niet bij een object.

   De **constructor** roept enkel de **setter** op om de naam in te stellen.

   De **getters** van de attributen geven de bijhorende waarde terug. Er is enkel een **setter** voor de naam en daarbij is geen controle nodig.

   De methode **geefNieuwePositie** berekent op basis van een dobbelsteenworp wat de nieuwe positie van de Speler op het spelbord wordt. Hierbij wordt het nieuwe vaknr ingesteld en de methode geeft een tekst terug die aangeeft wat er is gebeurd.

   Volgende mogelijkheden zijn aanwezig:

Positie	omschrijving	Resultaat
6	Brug	naar vak 12
42	doornstruik	naar vak 37
58	Dood	naar vak 1

   Als je voorbij het laatste vakje bent, wordt er teruggeteld. Bijvoorbeeld: (met GROOTSTENR = 63) als je op vak 61 staat en je gooit 5, dan kom je achtereenvolgens op 62-63-62-61-60 en is je nieuwe positie dus 60.

   De methode **gewonnen** bepaalt of de Speler zich op de juiste positie op het spelbord bevindt om te winnen. Het vaknummer van de Speler moet dus gelijk zijn aan het grootst mogelijke vaknummer op het spelbord.

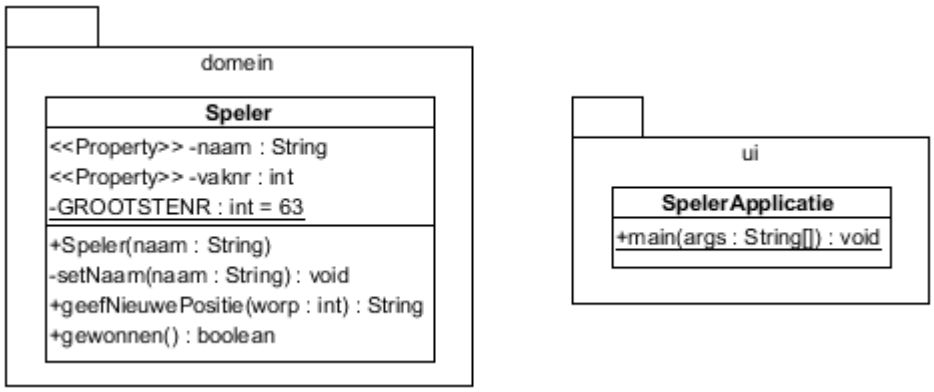
   De **applicatieklasse SpelerApplicatie** is een eenvoudige versie van het ganzenspel, waarin 3 methodes aanwezig zijn.

   De static methode **main** maakt eerst 2 Speler-objecten aan. Hiervoor wordt telkens de naam van een Speler ingelezen zodat deze aan de constructor van de domeinklasse kan doorgegeven worden.

   Daarna wordt at random een nummertje getrokken om te bepalen welke speler (1 of 2) mag beginnen.

   Vervolgens wordt tot het einde van het spel telkens de huidige speler bepaald (aan de hand van het nummertje kies je het juiste Speler-object), waarna deze speler een spelbeurt krijgt. In deze beurt zal de Speler de dobbelsteen laten rollen en vervolgens zijn pion naar het nieuwe vak verschuiven. We tonen hoeveel ogen de dobbelsteen aangaf en op welk vak de pion nu staat. Ook controleren we of de Speler gewonnen is en indien dit zo is, dan wordt hier ook melding van gemaakt en wordt aangegeven dat het spel ten einde is. Is het spel nog niet aan het eind gekomen, dan wordt het nummer van de speler die als volgende aan de beurt is, bepaald.

UML:



Gewenste in- en uitvoer:

Console

<terminated> SpelerApplicatie [Java Application] C:\Program Files\Java\jdk-11.0.4\bin\javaw.exe (15 sep. 2019 18:53:57)

```
Geef de naam van speler 1: Jan
Geef de naam van speler 2: Piet
Jan: je gooide 2 en je bevindt je op vak 2
Piet: je gooide 1 en je bevindt je op vak 1
Jan: je gooide 4 en je ging over de brug van vak 6 naar vak 12
Piet: je gooide 5 en je ging over de brug van vak 6 naar vak 12
Jan: je gooide 3 en je bevindt je op vak 15
Piet: je gooide 5 en je bevindt je op vak 17
Jan: je gooide 3 en je bevindt je op vak 18
Piet: je gooide 5 en je bevindt je op vak 22
Jan: je gooide 5 en je bevindt je op vak 23
Piet: je gooide 4 en je bevindt je op vak 26
Jan: je gooide 3 en je bevindt je op vak 26
Piet: je gooide 5 en je bevindt je op vak 31
Jan: je gooide 1 en je bevindt je op vak 27
Piet: je gooide 6 en je bevindt je op vak 37
Jan: je gooide 1 en je bevindt je op vak 28
Piet: je gooide 1 en je bevindt je op vak 38
Jan: je gooide 6 en je bevindt je op vak 34
Piet: je gooide 2 en je bevindt je op vak 40
Jan: je gooide 4 en je bevindt je op vak 38
Piet: je gooide 3 en je bevindt je op vak 43
Jan: je gooide 6 en je bevindt je op vak 44
Piet: je gooide 3 en je bevindt je op vak 46
Jan: je gooide 2 en je bevindt je op vak 46
Piet: je gooide 6 en je bevindt je op vak 52
Jan: je gooide 2 en je bevindt je op vak 48
Piet: je gooide 4 en je bevindt je op vak 56
Jan: je gooide 3 en je bevindt je op vak 51
Piet: je gooide 6 en je bevindt je op vak 62
Jan: je gooide 5 en je bevindt je op vak 56
Piet: je gooide 1 en je bevindt je op vak 63
Piet: je hebt het einde van het spel bereikt! GEWONNEN!
```